A METHODOLOGY FOR CONDUCTING DESIGN TRADES RELATED TO ADVANCED IN-SPACE ASSEMBLY

A Thesis
Presented to
The Academic Faculty

By

Lourenço Jara de Carvalho Vale de Almeida

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in the
School of Aerospace Engineering

Georgia Institute of Technology

December 2018

A METHODOLOGY FOR CONDUCTING DESIGN TRADES RELATED TO ADVANCED IN-SPACE ASSEMBLY

Approved by:

Prof. Dimitri Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Bradford Robertson
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Alexia Payan
School of Electrical Engineering
Georgia Institute of Technology

Date Approved: December 6, 2018

There are countless suns and countless Earths all rotating around their suns in exactly the same way as the seven planets of our system.

Giordano Bruno, 1584.

To Alana.

ACKNOWLEDGEMENTS

I would like to thank above all my advisor, Professor Dimitri Mavris, for offering me the opportunity to join the Aerospace Systems Design Lab (ASDL). ASDL is truly a world class institution. Over the past year I have learned and developed entirely new skills and applied them successfully on many different projects. Professor Mavris has always pushed us to aim for the highest standards, and his words of encouragement were invaluable in making this work possible.

I would also like to thank Dr. Bradford Robertson for the incredible amount of his valuable time he dedicated to helping me on this project over the past year, and for answering my endless series of questions. I also thank Dr. Payan for her valuable feedback on this and other projects.

Finally, I would like to thank my parents for making this journey possible in the first place.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In the decades since the end of the Apollo program, crewed space missions have been confined to Low Earth Orbit. Today, ambitious private and public efforts are underway to return astronauts to the surface of the Moon, and eventually reach Mars. Despite previous accomplishments, returning humans to deep space is difficult. Technical challenges and hazards to crew health and well-being posed by the deep space environment will require innovative solutions. The use of In-Space Assembly can provide critical new capabilities, by freeing designs from the size constraints of launch vehicles.

In-Space Assembly can be performed using various assembly strategies. The current state-of-the-art strategy, used notably for the assembly of the International Space Station, is to simply dock large modules together. New technologies, such as in-space welding, will make advanced strategies available in the mid-term future. Advanced strategies deliver small component pieces or materials to orbit in highly efficient packaging, and require complex assembly tasks to be performed in space.

The choice of assembly strategy impacts the cost and duration of the entire mission. Simpler strategies require more deliveries, increasing costs, while advanced strategies require more assembly tasks, increasing time. For the assembly of large space systems, different strategies may be used for different pieces. In addition, existing delivery and assembly capabilities may drive the choice of strategy. Alternatively, these capabilities may need to be sized for a specific mission. To design systems capable of meeting ambitious space exploration goals, the capability to conduct these design trades is needed.

A methodology to conduct these design trades is proposed. It uses a mathematical model of the complete assembly logistics of a space system, including the scheduling of deliveries and the scheduling of assembly tasks. The model is formulated as a network flow problem. Space systems are divided into structural sections, for which a network is created. In each of these networks, the pieces of a structural section must flow from their initial state to a final assembly state, via arcs representing deliveries and assembly tasks. These networks are linked by constraints representing assembly capabilities. Solutions can be obtained under different parameters, allowing design trades to be conducted.

This methodology is applied to the case of an Artificial Gravity Space Station, a structure that requires many deliveries to be assembled, but has the unique ability to mitigate the effects of microgravity. Results for the assembly of this system were obtained for different scenarios, by varying the parameters defining the available delivery and assembly capabilities and the desired objective. The comparison of the various results reveals the sensitives of assembly process to each parameter, and demonstrates the effectiveness of the proposed methodology.

CHAPTER 1

INTRODUCTION

1.1 Motivation - Human Space Exploration

1.1.1 History

The first human to reach space was the Soviet cosmonaut Yuri Gagarin, who orbited the Earth in April of 1961. Although Gagarin's voyage lasted less than two hours, this event marked the beginning of the Space Race, a decade of ambitious manned space exploration. By the following month, the United States began to catch up, when Alan Shepard became the first American to reach space. The next year, John Glenn became the first to reach orbit. After years of development and testing, and despite the disaster of Apollo 1, during which three astronauts perished, the United States persevered and eventually succeeded in landing Neil Armstrong and Buzz Aldrin on the surface Moon with Apollo 11, on July 1969. Five additional landings quickly followed, until Apollo 17 on December 1972, which marked the end of the Apollo Program, and the last time humanity journeyed beyond Earth and its lower orbit.

In the decades since, the exploration of the solar system has continued: multiple rovers have explored the surface of Mars, and various probes have traveled to distant planets. However, manned missions have remained confined to Low Earth Orbit, such as with the Space Shuttle and the International Space Station. Although advances in robotic technology have made the practicality of crewed missions a subject of debate, humanity's strong desire to expand beyond Earth continues to drive interest in human space exploration. Ambitious private and public plans call for returning humans to deep space in the short-term future, stepping once again on the surface of the Moon, and eventually reaching the surface of Mars. Despite previous accomplishments, reaching these goals will still require overcoming multiple challenges.

1.1.2 Goals and Challenges

The United States have recently expressed a desire to ”lead the return of humans to the Moon for long-term exploration [1]”. The Apollo-era systems were designed for short sortie missions, staying at most three days on the surface of the Moon. Therefore, a return to the Moon for long-term presence will require the development of new human-rated space systems. First of all, launch vehicles and transit systems must transport crew and cargo safely from Earth to the desired destinations and back. As an option, orbiting systems can provide stopping and refueling posts at critical points along the journey. Both these systems must be large and comfortable enough to accommodate multiple crew over long periods of time. Next, landing and ascent systems must provide travel to and from the planetary body’s surface for large amounts of cargo and crew. Finally, surface systems must enable exploration and safe stay on the surface for extended periods of time.

Accomplishing these goals is difficult. For long duration missions, the prolonged confinement of the crew in a small space can become a serious issue. However, transit systems, orbital systems, and surface systems are limited in size by what can be carried aboard launch vehicles. Therefore, constructing large systems capable of comfortably housing multiple crew over long trips or stays presents a technical challenge. In addition, Entry, Descend, and Landing (EDL) on Mars has only been accomplished for small probes and rovers of limited sizes. Human crews and cargo would require safely landing much heavier payloads, requiring a comparable scaling up of existing EDL solutions, or entirely new approaches.

In addition, the deep space environment presents numerous challenges to crew health and safety. Radiation exposure and micro-meteorites can seriously endanger the lives of the astronauts. Long term exposure to micro-gravity is another hazard that can affect the health of astronauts during the mission, as well as their transition back to a normal environment.

After spending a year aboard the International Space Station, the astronaut Scott Kelly suffered severe side-effects upon his return to Earth [2]. Such side-effects would have a negative impact on the ability of an astronaut to perform his duties upon arriving at his destination. Without tackling these various challenges, accomplishing the goal of returning humans to deep space becomes impossible.

1.1.3 Benefits of In-Space Assembly

Innovative designs can offer solutions to many of these challenges. Entry, Descent, and Landing (EDL) systems used for robotic missions to Mars benefit greatly from the use of atmospheric deceleration to reduce propellant mass requirements. However, very large heat shields would be required at the scale of payloads needed for human exploration. Rather than dealing with the difficulties associated with complex deployable and/or inflatable systems, such heat shields could be assembled in space from smaller component parts, for example by positioning shielding material on top of a backplane truss.

Similarly, artificial gravity systems have been proposed to mitigate the effects of long exposure to micro-gravity in transit vehicles or space stations. They consist of large spinning centrifuges that create a centripetal acceleration, simulating the effect of Earth's gravity to crew standing on the outer radius. To generate enough artificial gravity while mitigating certain side-effects, and provide enough room for a crew, such systems would be larger than any realistic launch vehicle. To be constructed, they would have to be built from parts delivered across multiple launches, and assembled into a geometry much different than any existing space system.

Other challenges and missions can benefit from the use of innovative designs assembled after launch. Surface systems, such as habitation modules, could be assembled on the ground from modular components delivered as simpler parts, rather than having to solve the problem of EDL with large payloads. Beyond human exploration, telescope mirrors could be mounted in space, avoiding extremely complex unfolding mechanisms. Similarly to heat shields, solar panels and antenna dishes could be assembled from component parts in space rather than limited by single payload capacities and deployment systems.

The ability to assemble structures in space can open up the design space beyond the severe restrictions of launch vehicle payload fairings, unlocking designs capable of overcoming multiple challenges. Technological limits on the size of launch vehicles, as well as safety concerns about their launch, in particular near population centers, restrict how large a rocket can be constructed, limiting payloads to a maximum size [3]. Although In-Space Assembly is itself a challenge and always comes at a cost, it has the potential to allow the return of humanity to deep space.

1.2 Background - In-Space Assembly

1.2.1 Overview of In-Space Assembly

In-Space Assembly (ISA) as a whole encompasses a set of different strategies used to assemble structures, vehicles, or any other space systems from assembly components, pieces, or parts [4]. A strategy represents a choice between how much work is to be done on Earth versus how much work is to be done in orbit. These strategies can be ranked by the complexity and the number of assembly tasks required to be done in orbit, which generally corresponds to smaller, simpler assembly components being delivered from Earth. Figure 1.1 (Adapted from [4]) shows the possible assembly strategies and how they are ranked in this scale.

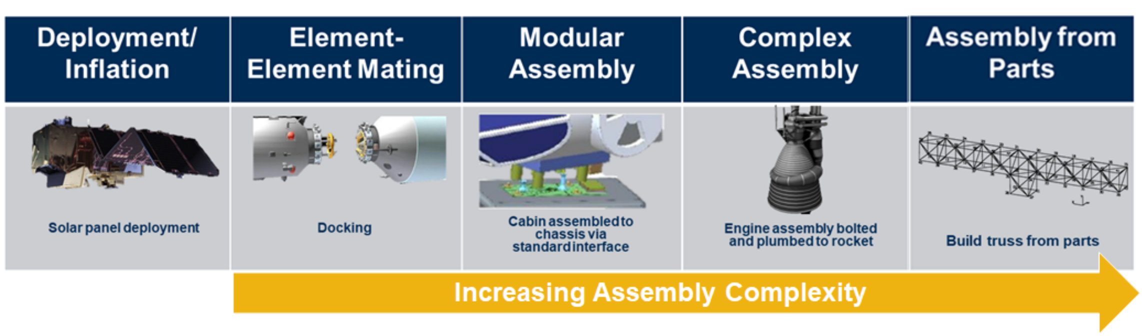

Figure 1.1: In-Space assembly strategies ranked by task complexity.

The vast majority of space systems, such as communication satellites, are usually deployed as a single pre-assembled unit, representing the ”level zero” of ISA. Despite the lack of actual assembly taking place in orbit, these systems can involve extremely complex deployment phases where different subsystems, such as solar panels, antennas, and dishes must unfold from their payload configuration. The first real ISA strategy, element-to-element mating, in which multiple modules are docked together to form a larger system, represents the most common ISA method, and has seen extensive use in the assembly of the International Space Station (ISS).

The next two advanced strategies beyond this state-of the art, modular and complex assembly, represent increments in the number of assembly tasks required and the joining technologies needed. Modular assembly makes use of common interfaces allowing easy "plug-in" of different components. In complex assembly, these tasks require more work, such as data and power connections that must be individually performed. At the end of the spectrum, assembly from parts represents the assembly of a system from its simplest components, such as struts and nodes forming a truss and multiple trusses forming a larger structure.

1.2.2 State-of-the-Art and Limitations

ISA has been a subject of interest to the space community for some time. Already in the 1960s, the Apollo missions used an element-to-element assembly strategy in docking the Lunar Module with the Command and Service Module. In the 1980s, NASA conducted extensive testing on astronaut Extra-Vehicular Activity (EVA) for the assembly of trusses. A Space Shuttle mission in 1992 manually assembled a truss section in the cargo bay using EVA [5]. Multiple Space Shuttle missions repaired and upgraded the Hubble Space Telescope using ISA techniques. The assembly of the ISS consisted primarily of berthing of modules using the on-board robotic arm, and autonomous docking, with extensive use of EVA for more complex tasks.

Despite its basic assembly strategy, the ISS represents the state-of-the-art of ISA to date. In fact, the technologies required to reliably perform more complex assembly strategies without the need for costly EVA are still being developed. As a result of these current limitations, most deep space exploration concepts have similar designs, shapes, and sizes. This narrow design space has a limited ability to fully tackle some of the challenges of deep space human exploration. Answers to the problems of Mars EDL and micro-gravity exposure remain uncertain. However, advanced In-Space Assembly has the potential to offer innovative answers.

1.2.3 Advanced In-Space Assembly as an Enabler

Advanced In-Space Assembly capabilities beyond the current state-of-the-art of element-to-element mating have the potential to enhance multiple human space exploration missions. Increased modularity, repairability, upgradeability, and spreading out of cost and risk over more launches can improve mission that are already achievable without advanced ISA. More importantly, progress along the ISA spectrum can enable new designs by eliminating the restrictions inherent to fairing-sized building blocks, making large systems, such as artificial gravity centrifuges, larger space telescopes, and human-scale Mars EDL systems possible.

A number of studies have highlighted the sizeable number of potential benefits from further ISA development, and identified the technologies needed to implement these more advanced strategies. Belvin et al. surveyed possible applications of ISA for an Asteroid Redirect Vehicle, an Artificial Gravity Vehicle, a Space Dock, a High Definition Space Telescope, Surface In-Situ Resource Utilization, Solar Electric Power & Propulsion, Sun Shields, Star Shades, and Atmospheric Decelerators [6]. The authors found that all can have lower costs, as well as more resilience through the use of advanced ISA. Robotic structural assembly and additive manufacturing were identified as critical technologies to enable these missions. In addition, achieving the reliable assembly of low mass, high stiffness structures in space will require closing technology gaps in joining technologies and robotic manipulators.

Focusing on the applications to a Space Tug, a next generation Space Telescope, and a Mars Transit Vehicle, Dorsey et al. concluded that modular, reversible interfaces, autonomous robotic assembly systems and verification in space are critical to successful development of ISA [7]. Similarly, Jefferies et al. identified five focus areas where applying ISA could support human missions to Mars: an Earth Space-Yard, a Mars Orbital Spaceyard, the Mars Transit Vehicle itself, Entry Descent and Landing systems, Surface Systems, and the communications infrastructure around Mars [4]. ISA was found to be capable of addressing the challenges of the human exploration of Mars by enlarging the design space and increasing the flexibility of the campaign architecture, enabling a broader range of usable launch vehicles, reducing risks and costs.

Hamill et al. used a two-step Quality Function Deployment methodology to identify the most relevant technologies to a set of ISA capabilities, and most relevant ISA capabilities to a set of relevant missions [8]. Again, the importance of high stiffness assembly and robotic assembly were highlighted. Commercial applications of ISA have also been investigated, in particular in the area of communication satellites. Their primary performance, in terms of bandwidth, comes from the aperture of the main reflector, typically limited by launch fairing size limitations. A study found that incorporating ISA can double the bandwidth and therefore greatly benefit communication satellites [5].

The potential of advanced ISA to tackle multiple challenges, and in particular those related to human space exploration, is clear. The critical technologies needed to reach the necessary capabilities have also been identified and are being developed. Assuming a steady rate of progress, advanced ISA strategies will be available to the space community in the mid-term future. However, their implementation will come with multiple trade-offs that will have to be considered by the system, mission, and campaign designers.

1.3 Research Objective

1.3.1 Advanced In-Space Assembly Design Trades

In-Space Assembly has been used to accomplish previous human space exploration goals, and will be used again for future missions. However, technological developments will soon make advanced ISA strategies available to the mission planner alongside the current state-of-the-art techniques. The choice of ISA strategy will impact the mission as a whole. Simpler strategies will require more launches to deliver the same components to orbit, increasing the number of deliveries and their costs. However, the assembly processes required will remain relatively short and simple, with fewer investments in ISA capability being needed. More complex assembly strategies can be implemented to reduce the amount of deliveries used and the launch costs associated with each, by increasing the packaging efficiency of each payload. However, they will require larger investments in space assembly infrastructure and capabilities, as well as necessitate longer, riskier, and costlier assembly processes in orbit.

In addition, different assembly strategies can be combined in the construction of a single vehicle, system, or overall campaign. A truss section part of a larger space system can be assembled from component pieces, while an adjoining habitat is docked to another using current element-to-element mating methods. Two different trusses on the same system may be assembled using entirely different strategies, depending on the assembly and delivery capabilities available at the moment. The choice of strategy can be a piece-by-piece decision dependent on the available options at the time of assembly. It is also reasonable to assume that future delivery and ISA capabilities may not be sized or dependent on a single mission or assembly project. The development of reusable launch vehicles may usher an era of frequent and cheap space cargo. A complex ISA infrastructure such as a space-yard may be developed to serve other needs, and exist as the context of a new project. The correct choice of ISA strategy would then correspond to the best use of those existing capabilities.

With ambitious human space exploration goals, numerous technical challenges and hazards to crew health and safety, and new technologies enabling the assembly of large, complex structures capable of tackling those challenges, new design trades will need to be conducted while designing future space systems in order to optimize overall mission metrics such as costs and required assembly time. The **Research Objective** of this work is therefore to **develop a methodology that allows the conducting of these new design trades involving the choice of delivery configuration, the arrangement and scheduling of deliveries, and the scheduling of assembly tasks**. Examples of these trades include:

- What assembly strategy should be used for a space system?
  - Simpler strategies require more deliveries (Cost), more complex strategies require more assembly tasks (Time)
  - If the assembly capabilities and/or the delivery capabilities are already sized, how can they best be used?
- What assembly strategy should be used for different parts of a space system?
  - Different strategies can be combined to assemble a single system
  - The use of a strategy for a part of a system may interact with the availability of that same strategy for another part of the system. How can they be best combined?
- How should In-Space Assembly capabilities be sized?
  - With the goal of assembling a space structure within a set time or budget, and a given delivery capability, what benefits do additional assembly capabilities provide, and are they worth it?
- How should Delivery capabilities be sized?

- With the goal of assembling a space structure within a set time or budget, and given assembly capabilities, what benefits do additional deliveries provide, and are they worth it?

1.3.2 Research Question and Thesis Outline

In order to achieve the research objective of performing design trades related to advanced ISA, a single **Research Question** is asked:

What is the correct method of conducting design trades related to advanced In-Space Assembly strategies?

This research question is answered in Chapter 2 by proposing a method of conducting design trades that uses a mathematical model of ISA logistics. This methodology and the accompanying mathematical model developed using flow networks and linear programming are described in detail in Chapter 3. Finally, the methodology is applied to the case of the assembly of an Artificial Gravity Space Station in Chapter 4, in order to showcase various design trades related to ISA.

CHAPTER 2
IN-SPACE ASSEMBLY AS A LOGISTICS PROBLEM

2.1 Complexity of the ISA Problem

Although very advanced from a technical perspective, the problem of docking two separate modules to form a larger spacecraft can be considered relatively trivial from a logistics point-of-view. Each module is sized to maximize the available payload capacity, and delivered separately on different launch vehicles. A single sequence of assembly tasks is required, and this sequence begins as soon as the modules are in proximity. The number of launches and the time needed to complete the assembly are fixed. There are no degrees of freedom to manipulate in order to obtain a better outcome, and there are no design trades that can be conducted for the assembly process.

The problem is different when the space system to be assembled consists of many pieces. Each piece must be allocated to a delivery for transport from its origin to the assembly orbit, according to its size and the delivery capabilities. Multiple assembly strategies may be available for each piece, and as such, the number and size of each piece must be determined considering both delivery availability and capacity. In addition, pieces tightly packed together may arrive in a single delivery, but will require lengthy assembly processes to achieve the desired arrangement. Because the assembly tasks required for each piece must be scheduled according to the capacity of each assembly system, these various aspects of the problem are dependent on each other. With more pieces, more assembly tasks will be required. If the capability to perform a certain task is saturated (meaning that the piece must wait before it can undergo said task process), the use of a certain assembly strategy may be temporarily unavailable. Similarly, a long gap between deliveries may favor certain assembly strategies.

Logistics is defined as “the detailed coordination and implementation of a complex operation” [9]. Because of the complex problems involved, and because all must be tackled simultaneously, assembling a large space structure in space is a logistics problem. In fact, it is a *space logistics* problem, defined as ”the science of planning and carrying out the movement of humans and materiel to, from, and within space combined with the ability to maintain human and robotics operations within space” [10]. In this space logistics problem, the number and size of pieces, their delivery allocation, the schedule of deliveries, and the schedule of assembly must all be determined at the same time.

2.2 Review of Relevant Problems and Methodologies

Although the field of space logistics is relatively recent, a number of efforts have been conducted to develop models to a variety of these problems. An overview of developments on the topic can be found in [11]. A good amount of space logistics research has been applied to campaign design. Campaigns consists of multiple missions carried over a period of time towards a common goal with shared architectural elements. The Apollo program took a “carry-along” approach: each crew traveled with all its needs on each mission. Future missions, in particular to Mars, will have to make use of resupplying and maybe even in-situ resources [12]. Therefore, the questions of what to send, use, extract, where and when, become an important degree of freedom that justifies the use of a logistics optimization approach.

Taylor et al. formulate the campaign logistics problem as a network flow, where commodities must flow between nodes representing different orbits and surface locations [13]. This problem is solved using Integer Programming (IP), and applied to a lunar sortie mission. This network flow approach is modified to serve broader applications than Lunar exploration by Ishimatsu et al, who expand the work of Taylor and apply it to the human exploration of a near earth object and Mars [12]. Once again, campaign logistics are modeled as a network flow problem. Destinations and orbits are the nodes of a mathematical graph. Variables representing the node flows of different commodities are subject to constraints in an Mixed Integer Linear Program (MILP) formulation. Integer variables are used as necessary, for example to avoid splitting a single crew member between multiple missions, or to model discrete events such as individual launches. Network nodes can provide a positive outflow when ISRU is used.

Ho et al. further develop this formulation by expanding the network in the time dimension: each node now exists at different times [14]. This removes time paradoxes from the previous static network formulation. The authors make this approach more computationally efficient using finer time steps only at critical time windows in [15]. Finally, the calculation of a solution is further simplified by keeping the network partially static at local clusters in [16]. The time dimension, a large source of computational complexity, is only maintained where absolutely necessary (between cis-lunar nodes and Mars nodes for example) and removed entirely where doing so constitutes an acceptable simplification. This campaign-level approach is further combined with elements of spacecraft and space infrastructure design by Chen and Ho, in order to concurrently optimize missions, vehicles, and fuel depot design [17]. To facilitate computation of a solution, quadratic terms in the spacecraft design model are approximated by piecewise linear functions and the problem is turned into MILP. Another approach, simulated annealing, is proposed to allow for higher fidelity spacecraft models, at the cost of less robust results.

Problems involving the logistics of space stations tend to look at the ”sustain” phase of the mission, not the assembly part. Ho, Greem and de Weck propose and refine a model for the combined design of both a space station and its logistics [18] [19] [20]. The objective of minimizing cost, a function of the launch capacity and frequency, is coupled with the problem of maximizing scientific return, a function of the number of crew and maintenance level, but also of the launch frequency and capacity. The goal is to find the optimal combination of these variables.

Lin, Luo, and Tang develop another space station logistics optimization formulation, this time considering both the launch times of resupply vehicles and their cargo manifests as design variables [21]. Optimization of the utilization and robustness of the system is achieved using a genetic algorithm. In [22], they develop a multi-objective optimization to maximize the space station crew stay duration, and minimize launch costs, among others, by varying the crew complement, rotation, and launch capacity and number, using a genetic algorithm once again to solve the complex Mixed Integer Non-Linear Program (MINLP). Finally in [23], the same authors propose a decomposition approach to expands the space station logistics optimization to cover all aspects of the problem: mission planning, vehicle visit planning, and orbital flying planning. In another study, Fasano also develops a model of the sustainment-phase logistics for a space station [24]. There it is modeled as warehouse problem, with different capacities and needs. Among the various commodities considered are resource supplies, but also maintenance, reboosting, and return to earth of experimental material. The model considers different launchers and vehicles, with different capacities and delivery frequencies. The objective is to minimize costs.

Although space station problems tend to look at sustainment phases only, considering assembly trades is not unheard of for other applications. Morgenthaler offers an example of a design optimization based on space logistics considerations, where the total expected cost of the assembly of a Mars transit vehicle is modeled as a function of a launch vehicle mass to orbit capability [3]. In order to size the optimal launch vehicle, the model accounts for the cost of the launch vehicle, its mass and payload capability to determine the number of launches required, the cost and number of assembly and connection tasks required, and the costs of launch failure. Similarly, Moses et al., propose a Life-Cycle Cost Analysis methodology to identify the impact of modularity on space missions [25]. This approach highlights how the choice of ISA strategy can effect the mission as a whole by impacting not only the design of a system but also its launch and assembly costs.

In another application, Sternberg et al. present an optimization for ISA of a modular satellite using a genetic algorithm [26]. The objective of this research is to minimize downtime and cost of assembly by varying the number of assemblers, their position in orbit, their degree of autonomy, and their assembling capacity. In [27] and [28], Gralla and De Weck compare different assembly strategies, from self assembly to the use of a single or multiple tugs, with the goal of finding the optimal method of assembly of modular Earth satellites. These methods are particular for considering orbital mechanics as an aspect of the problem.

Other space logistics problems offer relevant insights to the problem of ISA logistics. In [29], Mixed Integer Programming (MIP) is used to solve the optimization of a planetary surface exploration campaign, and applied to the exploration of the surface Mars. The goal is to select which sites should be visited and by which method (walk or rover), in order to maximize the value obtained while staying within budget. The value of a certain technology (in this case, supplying exploration vehicles using orbital depots) is obtained by comparing the solutions obtained with and without the routing strategies enabled by that technology. Using a model under different scenarios is a common technique of conducting design trades.

In [30], Fasano solves the problem of resource-constrained task scheduling, such as different payloads on an spacecraft with variable power capacity, using a MILP optimization. Time is discretized into time-units in order to transform the complex continuous time problem into a more easily solvable time-indexed problem. Both Fasano and Gliozzi also apply MILP approach to the problem of storage in a space module. The goal is to maximize the volume exploitation, while obeying various packing constraints [31] [32]. In [33] and [34], Siddiqi and Grogan propose a matrix formulation of space cargo manifest. Combined with MILP optimization of delivery logistics, this method allows efficient representation of the delivery allocation of pieces.

While the flow of commodities, the assignment of tasks, or the optimization of a design are more common space logistics problems, the problem of division of a designed structure into assembly parts is more common in ship-building. Methods for the optimization of block division (by minimizing weld length) under size and weight constraints have been implemented using genetic algorithms [35] [36]. The problem of ship hull division, in which the size of plates must be determined in order to optimize assembly time and cost, has close analogies to future advanced In-Space Assembly trades. Larger hull plates require more forming time but less welding time, while smaller hull plates require less forming time but more welding time. An optimization approach for this problem, using differential evolution, is developed in [37].

2.3 Choice of Methodology and Hypothesis

Although campaign logistics and space station resupplying optimization methods can offer answers to the questions of delivery allocation and scheduling, they do not specifically consider assembly tasks. Modular spacecraft assembly optimization methods tackle design and assembly capacity trades, but only consider small scale problems with a handful of pieces, not the kind of systems for which advanced ISA will be required. The choice of division and pre-assembly strategy for a part finds its closest analogues in the problems of hull and block division in ship building, a far cry from space exploration, and do not consider the linking problem of delivery. As of yet, no study has attempted to tackle the combined problems of advanced In-Space Assembly logistics for a large structure. It is hoped that the work of this thesis will fill that gap.

The review of current efforts in space logistics reveals the widespread use and effectiveness of mathematical models. Such models allow the comparison of various scenarios in order to determine sensitivities, ideal parameters, and conduct design trades. To properly conduct the desired design trades for the problem of In-Space Assembly, a new method is proposed. This method develops a new mathematical optimization model of the complete ISA logistics, considering both deliveries and assembly tasks, in order to calculate an optimal schedule of the entire assembly process of a space system. With the model fully operational, various scenarios can be run and additional design trades conducted by comparing their results.

The conceptualization of complex problems as network flows results in the relatively straightforward formulation of mathematical programs. With modern computers and commercial optimization solvers, such mathematical programs can be reliably solved to near-optimality, bypassing the need for heuristics such as genetic algorithms, which are furthermore not adapted to the complex constraints involved with logistics scheduling problems. A model of ISA logistics can harness the effectiveness of the network flow approach to generate an easily solvable mathematical program.

The research question can now be answered in two parts:

What is the correct method of conducting design trades related to advanced In-Space Assembly strategies?

- **Overarching Hypothesis**: The correct method of conducting these new design trades tackles advanced ISA as a combined **logistics** problem. It uses a model of In-Space Assembly logistics as a whole, in which the delivery configuration, delivery schedule, and assembly sequence are all degrees of freedom for the optimization of the entire assembly process. In addition to the trade-offs performed automatically by the logistics model, additional trades can be performed from the comparison of results obtained under different scenarios. This methodology allows design trades that are impossible when taking the different problems related to ISA separetely.

- **Sub-Hypothesis**: An effective model of ISA logistics can be developed using a network flow approach. Network flows allow the formulation of mathematical programs with a minimal number of variables and constraints. In the field of space logistics, minimum cost flow problems are commonly formulated from time-expanded networks. Such an approach can be adapted to the problem of ISA logistics to provide an optimal or near-optimal answer to the choice of assembly strategy, delivery schedule and assembly sequence.

CHAPTER 3

METHODOLOGY AND LOGISTICS MODEL DESCRIPTION

3.1 Methodology Overview

A mathematical model of ISA logistics forms the core of the methodology described in this chapter. This model is used to calculate the best possible solution to the assembly of a space structure for a given input scenario. For each scenario, the best possible solution is one that minimizes or maximizes the desired parameters, such as the assembly time or delivery costs. Once obtained, solutions under different scenarios can be compared, allowing design trades related to advanced ISA to be performed. A flowchart of this methodology is shown in Figure 3.1.

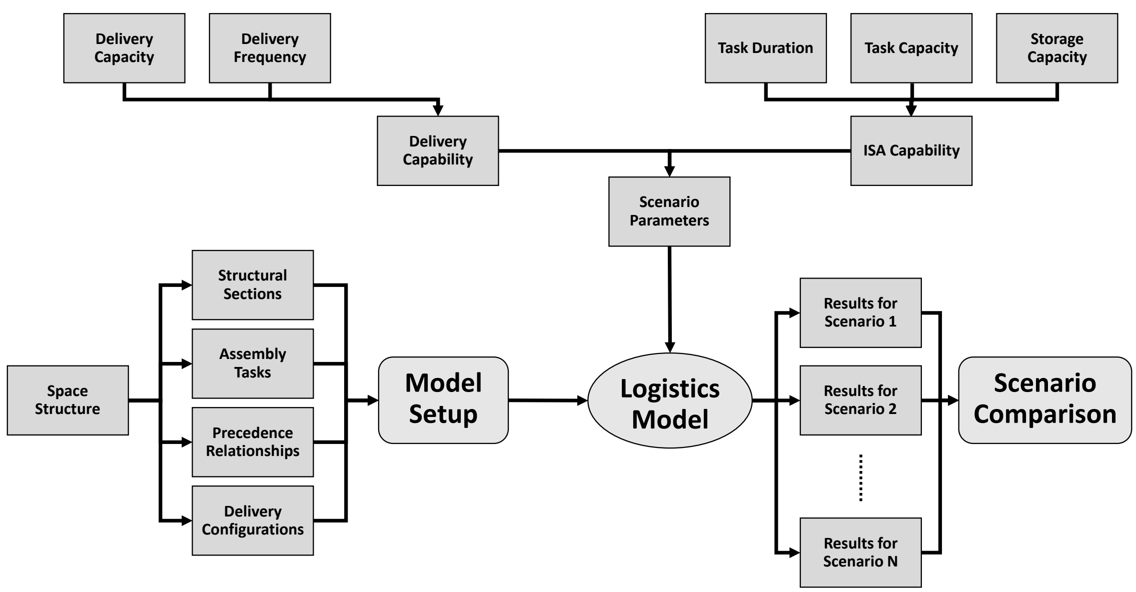

Figure 3.1: Methodology Overview

To enable trades for a variety of relevant cases, the model is designed to be flexible to the assembly of different space systems and structures, such as space stations, Entry-

Descent-Landing (EDL) systems, space telescopes, or even surface colonies. Before obtaining solutions for different scenarios, the logistics model must be setup for the space structure of interest.

A space structure is first represented as an arrangement of multiple **structural sections**. These structural sections can be of one of two types, defined by the way in which they can be divided into component **pieces**. The first type represents structural sections that can be divided into a **fixed number of interchangeable pieces of equal size**. Trusses are an example of this type of structural section: they can be divided into "pieces" consisting of multiple struts arranged to form a pattern. Each pattern is interchangeable with the next, and the number of patterns required to form a truss is fixed by the dimensions of the truss. For this type of structural section, an interesting ISA trade involves the choice of **delivery configuration**. Delivery configurations represent different ways to transport a piece to the assembly location, each with their advantages and disadvantages. In the example of a truss, at least two delivery configurations are available. Truss pieces can be delivered in flat-packed or pre-assembled configurations. The flat-packed configuration offers more packaging efficiency at the cost of assembly time. The pre-assembled configuration offers faster assembly at the cost of packaging efficiency.

The second type represents a structural section that can be divided into a **variable number of pieces of variable sizes**. Just like the first type, these sections have a fixed dimension, such as length, area, or mass. However, they can be divided into pieces of variable sizes, depending only on the delivery capacity available. Space habitats, heat shields, or telescope mirror surfaces are examples of this type of structural section. Space habitats can be divided into a variable number of pieces of variable lengths to achieve a fixed total length. Similarly, heat shields can be divided into a variable number of pieces of variable areas to achieve a total surface.

The assembly process for each structural section is modeled with **states and tasks**. States represent the different steps required between the initial delivery and completing the assembly of a piece. To advance from one state to the next, an assembly task must be performed. For example, a truss piece in the "pre-assembled" state will require an "assembly" task to be performed in order to reach the "assembled" state. Depending on the available data and the desired accuracy of the model, the assembly of a piece can be modeled with any number of different tasks of different durations. The assembly of a truss could be represented as a set of tasks for each individual strut placement, or as a single task encompassing every required step.

Pieces and their tasks may be linked by **precedence relationships**, which relate the assembly of a piece to that of another. In the truss example, each piece within that structural section can only be assembled once the previous piece has been completed. In the case of a space station, a habitat section cannot be started until the corresponding truss structural section is finished. Once again, the complexity of these precedence relationships depends on the desired accuracy of the model.

With the logistics model configured for a particular space structure, solutions to different scenarios can be calculated. An input scenario consists of defined **delivery and assembly capabilities**. The delivery capability is represented by delivery frequencies (how often a delivery of a certain type occurs) and by delivery capacities (how much can be carried on a delivery of a certain type). Different delivery types represent different systems with independent characteristics. By reducing the delivery capability to these two simple parameters, the model remains flexible to different concepts of operations, from conventional launch vehicles to space tugs.

The assembly capability is represented by task durations, task capacities, and storage capacities. Task durations represent the amount of time required to perform each assembly task. Different assembly tasks may have different durations. The capacity of a task is the ability to perform the same task simultaneously on different pieces. This capacity can represent multiple systems operating together or a single system capable of performing multiple tasks at the same time. To fully define a scenario, the model also requires storage capacities, which represent how many pieces can be stored at each state (between assembly tasks). These storage capacities indicate not only "pure" storage, such as pieces awaiting assembly, but also the capacity to hold pieces in intermediate states, such as assembled trusses waiting to be inspected and verified.

Once obtained, the solutions to each scenario provide a schedule of deliveries, the payload manifest of each delivery, and the timeline of assembly, including when each piece undergoes each assembly task. Different scenarios can be compared in terms of the total cost and/or number of deliveries, the amount of time taken for the complete assembly, or other measures such as the downtime of different tasks or the amount of time spent in storage. In this way, the impact of the different parameters on the overall assembly process can be assessed.

3.2 Linear Programming and Network Flows

The purpose of the logistics model is to provide the best possible solution for a particular scenario, such that different scenarios may be compared on an equal basis. Providing the best possible solution requires optimization. Therefore, the logistics model is formulated as a mathematical program. A mathematical program seeks to solve a set of constraints on its variables while minimizing a cost function of the same variables. When these constraints and cost function are linear, the program is called a linear program. Solving large linear programs is made relatively easy by modern computer processors and commercial optimization solvers. The challenge lies in formulating a real-life problem, such as ISA logistics, as a linear program.

As part of a class project, a prototype formulation of the ISA logistics problem for the assembly of a space station was developed as a linear program. This formulation used a large number of binary variables to track the progress of each structural section piece, at each state, and at each discrete time increment. Additional variables tracked piece sizes and payload manifests, and a vast array of complex constraints linked all these together.

This initial formulation proved too unwieldy to properly conduct trades, due to its long solution times and its lack of flexibility. As a result, the search for a better formulation was necessary.

A network flow approach can offer an efficient formulation of a complex problem as a linear program. This approach enables an intuitive visualization of the required variables and constraints, while minimizing their complexity and number. A network, or mathematical graph G, consists of a set of nodes or vertices V and a set of arcs or edges E connecting those nodes together. A typical network flow formulation assigns variables x_{ij} to each arc in E, where (i, j) refers to the arc from node i to node j. Then, for each of the nodes in V, a linear constraint is created such that the sum of the flow coming into a node is equal to the sum of the flow leaving the node. Special nodes, called sources and sinks, can generate or capture flow, and have a non-zero "inflow" I_i. In addition, each arc is limited by its capacity, between l_{ij} and u_{ij}, where the lower limit is typically zero. The problem is solved when the flow is consistent across the network. By assigning costs c_{ij} to each arc, the minimum cost flow can be calculated. Equation 3.1 shows this basic formulation.

$$\begin{aligned} \text{minimize} \quad & \sum_{(i,j)\epsilon E} c_{ij}x_{ij} \\ \text{subject to} \quad & \sum_{k:(k,i)\epsilon E} x_{ki} + I_i = \sum_{j:(i,j)\epsilon E} x_{ij},\ \forall i\epsilon V \\ & l_{ij} \leq x_{ij} \leq u_{ij}, \forall (i,j)\epsilon E \end{aligned} \tag{3.1}$$

Network flows are often used to model the movement of commodities. A typical application of the minimum cost flow problem, where commodities must be moved from sources to sinks along the cheapest path, is the problem of the movement of the production from factories (sources) to different consumers (sinks), as shown in Figure 3.2. Different arcs represent different delivery methods, each with its own capacity and cost. Some arcs may not exist or be temporarily unavailable. The optimization seeks the optimal path to satisfy the demand of every consumer. Network flows can also be used for task assignment problems where the ”flow” on an arc no longer represents a commodity, but a logical decision. The other example shown in 3.2 must assign 4 tasks to 4 different workers. Each task must be assigned once and only once (the arcs to the sink have a capacity of 1), but it is possible that a worker is assigned to multiple tasks. Drawing the problem as a network and properly constraining the arcs allows the variables and equations to be automatically obtained. Due to their flexibility, flow networks can also be used for problems as complex as ISA logistics, with some modifications.

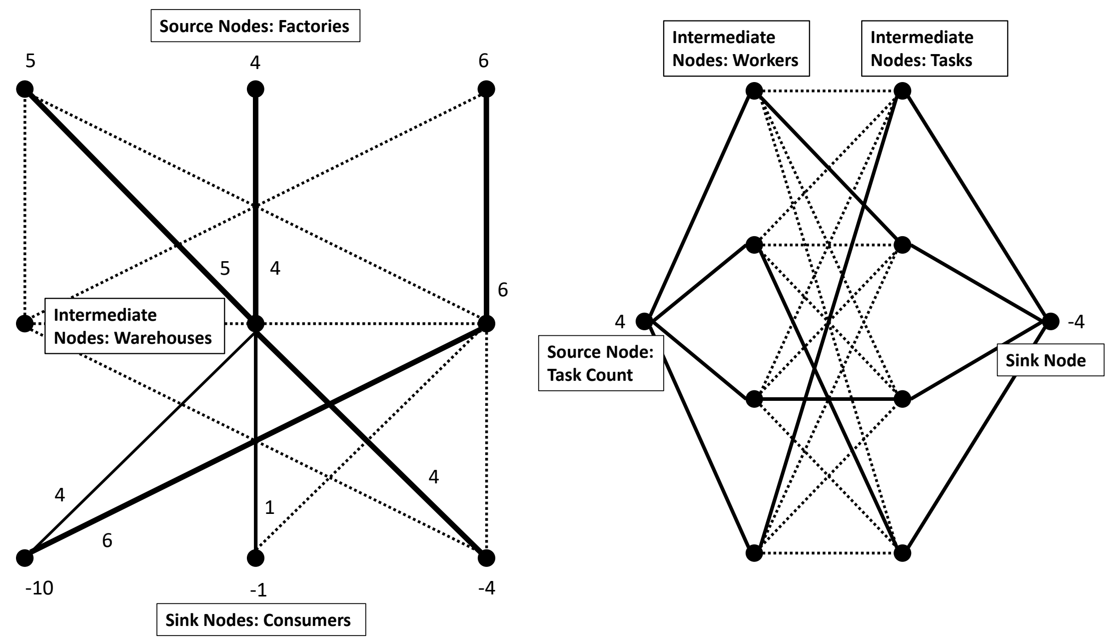

Figure 3.2: Example networks used to model transportation and task assignment problems.

3.3 Modeling Structural Sections with a Fixed Number of Pieces

3.3.1 Basic Network Flow Formulation

Network flows can be applied to the problem of ISA logistics. The process of assembling a structural section can be represented as a flow of pieces from a source to a sink, passing through various intermediate nodes. A few changes in how a typical network flow problem is understood must first take place. First, the different nodes must represent different states of assembly rather than just physical locations. A single source node is created to represent the origin of deliveries, typically Earth, as well as their initial state. A single sink node is created to represent the state of assembly being completed at the assembly location. Intermediate nodes are created to represent the various steps required to fully assemble each piece, between the source and the sink. Second, the network must be time-expanded, such that nodes exist for each state at each discrete time interval. This time-expansion allows the model to take into account delivery frequencies and the duration of states. The network shown in Figure 3.3 is an example of this approach.

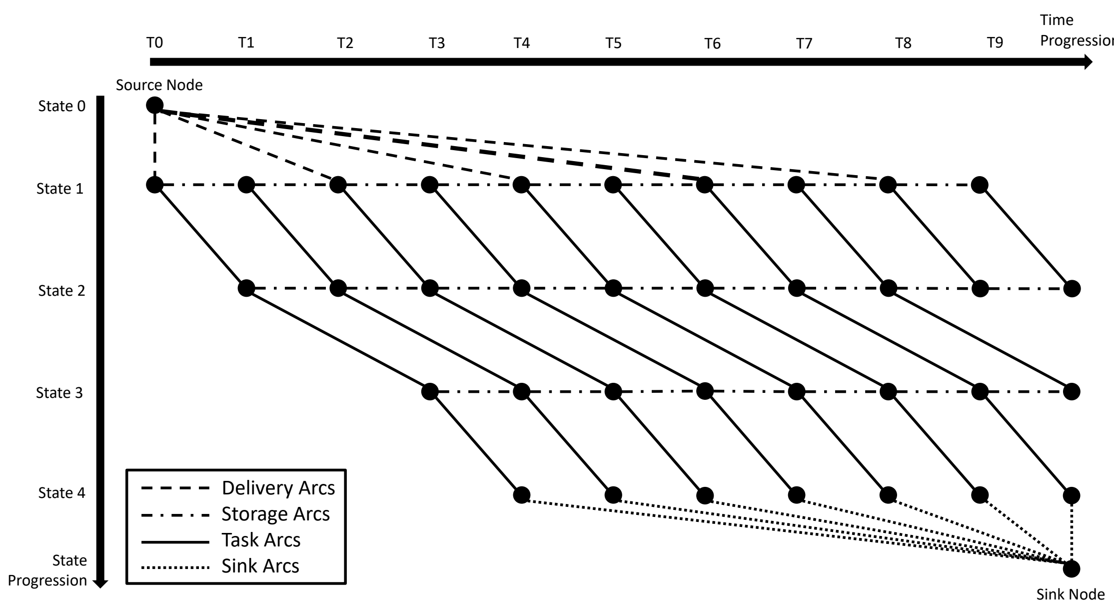

Figure 3.3: Basic time-expanded ISA network.

Nodes exist at each point in the "state-time" dimensions and can therefore be represented by an (S, T) pair where S is the state and T is the time of the node. In Figure 3.3, $S\epsilon\{S0, \ldots, S5\}$ and $T\epsilon\{T0, \ldots, T10\}$. The arcs connecting these nodes are specific to the space structure being assembled and the scenario being considered. There are four types of arcs. Delivery arcs connect the source node at $(S0, T0)$ (state 0, time 0) to $S1$ (state 1) at different T (time intervals). Note that not all $S1$ nodes are connected to the source. The existing arcs represent the available deliveries of the input scenario. Deliveries of different types are represented by different frequencies and different capacities on their respective arcs. The example of Figure 3.3 shows two delivery types, one with higher frequency and lower capacity, the other occurring only at $T6$ but with higher capacity (represented by the arc width). Although the source node is placed at $T0$, it is understood that deliveries occur at the time that their arcs connect with a $S1$ node. The source node takes an "inflow" value equal to the number of pieces requiring assembly.

Task arcs connect the various states of assembly between each other. Each node (S, T) is connected to the node $(S + 1, T + D)$, where D is the duration of the task required to transition between those two states. The duration of each task is represented by the slope of the task arcs in Figure 3.3. Different tasks can have different durations: in the example shown in the figure, the tasks $(S1 \rightarrow S2)$ and $(S3 \rightarrow S4)$ have duration of 1, while task $(S2 \rightarrow S3)$ has duration of 2 TU (time units). Both delivery frequencies and task durations are multiples of a chosen time discretization that defines the time interval between each node. Finer time intervals create larger networks which require more time to solve.

Storage arcs link nodes at the same state over time. They represent the passive process of maintaining a piece at its current state. These arcs have their own capacity that can be defined independently for each state. The final state (state 4 in the example shown in Figure 3.3) does not allow for storage arcs. Instead, it links directly to the sink via sink arcs, which simply "collect" the flow. Since the source node has an "inflow" equal to the number of pieces to be assembled, the sink must have the negative of this value for the flow to be conserved.

The linear program obtained from applying the flow conservation equation (3.1) to this network can be solved to obtain a basic solution for the assembly of a structural section. To model additional aspects of ISA, this formulation must be expanded with additional arcs and constraints.

3.3.2 Delivery Configurations and Delivery States

Since many ISA design trades are related to the choice of delivery configuration, the network must be able to model this degree of freedom. Therefore, different delivery configurations are implemented in the previously described network formulation by adding delivery arcs to states beyond the first assembly state, as shown in Figure 3.4. These arcs occur at the same frequency to each delivery configuration, and do not represent different physical events, but rather different ways to package pieces within the same delivery system. Arcs to more advanced states represent the packaging of piece closer to its final assembly state, allowing a piece to skip one or multiple assembly tasks. To properly reflect ISA trade-offs, such a shortcut must be accompanied by an associated penalty: the lower capacity of these delivery arcs, representing the additional payload size taken by pieces delivered in an advanced assembly state. Although more pieces can be delivered to a less advanced state using the same delivery event, those pieces require one or more additional tasks to "catch up." The simplest example of this type of trade-off is once again in the case of a truss structural section, between delivering the piece in a flat-packed or in a pre-assembled state.

Because these additional arcs do not represent new physical deliveries, the delivery arcs arriving at a particular time must be linked by a constraint. This constraint (Equation 3.2) is the first additional constraint to the model beyond the "pure" network flow conservation constraint. It ensures that the total capacity of a unique delivery event is not exceeded by the sum of the capacity used for each delivery configuration. In other words, for each delivery event, the sum of the relative capacity usage $\frac{x_{ij}}{K_{ij}}$ for each delivery configuration must be less or equal to 1, where K_{ij} is the delivery capacity between node i and node j. This constraint is applied at each time t when a delivery occurs, and sums the arcs connecting the source node to each node j in the set of nodes existing at time t and each state in DST, the possible Delivery STates corresponding to different delivery configurations. The operators $T(i)$ and $S(i)$ return the time and state of a node, respectively.

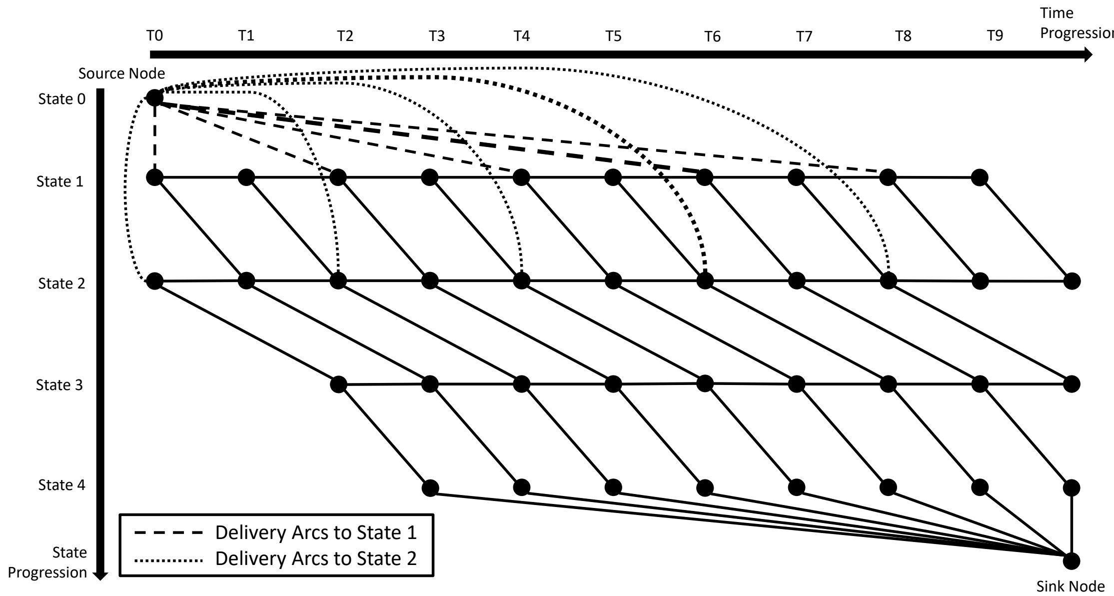

Figure 3.4: ISA network with additional arcs to model multiple delivery configurations.

$$\sum_{(i,j)\epsilon D_t} \frac{x_{ij}}{K_{ij}} \leq 1, \forall t \tag{3.2}$$
$$\text{where } D_t = \{(i,j)\epsilon E : (S(i) = 0, T(j) = t, S(j)\epsilon DST)\}$$

Considering an example of a structural section allowing two possible delivery configurations, and representing nodes using their (S,T) pairs, where the pair $(0,0)$ represents the source node, a delivery event with capacity $K_{(0,0)(1,t)} = 10$ for the first delivery configuration and $K_{(0,0)(2,t)} = 2$ for the second delivery configuration, then a possible payload could include 5 pieces in the first configuration along with one piece in the second configuration $(5/10 + 1/2 = 1)$. This constraint formulation allows the mixing of different delivery configurations among the pieces being delivered in a single event.

3.3.3 Precedence Relationships

The network formulation described so far does not allow for precedence constraints to be established. A truss piece can begin its final assembly while another is still being completed. Before adding precedence rules to the model, the assumptions made about the type of structural section being modeled must first be laid out. Structural sections modeled by an individual network consist of pieces that must be assembled one by one. Each piece is assembled “on top” of the previous piece. Each piece is interchangeable and can only be assembled either first, or after a previous piece in the same section has completed assembly. No pieces are assembled simultaneously. More details on how to divide a larger system into these “1-by-1” structural sections are given in the following sections.

Properly accounting for this “1-by-1” precedence relationship requires another set of constraints to be added to the basic network formulation. First, the capacity of each arc related to an assembly process for which precedence is required is limited to a maximum value of 1. Next, a “time cut” is performed at each time interval such that certain arcs indicating simultaneous tasks (shown in Figure 3.5) are exclusive with each other. The resulting constraints (Equation 3.3) refer to PST, the Precedence STates for which precedence is enforced. Not all states of assembly have to be constrained by precedence: for example, unpacking a flat-packed truss piece can be performed regardless of the assembly states of other pieces. However, assembling that piece to the rest of the truss arm requires all prior pieces to have been completed. The arcs that must be exclusive at each time t are the arcs belonging to the set P_t, consisting of the arcs beginning at each of the Precedence STates at or before time t that have not yet reached their destination at time t. Storage arcs at the very first precedence state are excluded from this constraint (a piece in “storage” at the first of the precedence states is merely waiting for that task to begin, and is free to do so while others are assembled).

$$\begin{aligned} & x_{ij} \leq 1, \ \forall (i,j) \epsilon P_t, \forall t \\ & \sum_{(i,j)\epsilon P_t} x_{ij} \leq 1, \ \forall t \\ & \text{where } P_t = \{(i,j)\epsilon E : (T(i) \leq t, T(j) > t, S(i) \epsilon PST, S(j) \neq min(PST))\} \end{aligned} \tag{3.3}$$

In this way, assembly tasks for the ”next piece” can only occur when no other piece is

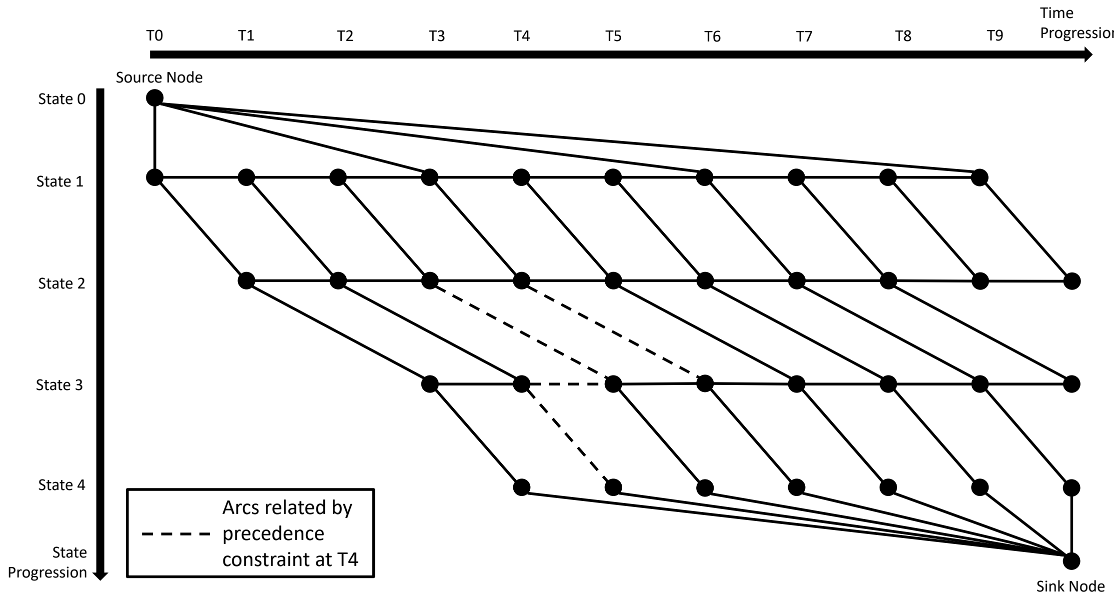

Figure 3.5: ISA Network showing arcs related by ”1-by-1” precedence constraint

being assembled. The practical result is that the assembly processes requiring precedence only begin after previous pieces have been completed. This solution allows the modeling of the simple “1-by-1” precedence in the efficient network flow formulation. Figure 3.6 displays a possible assembly sequence for a structural section consisting of two pieces, and for which states 2 and 3 are precedence states, but state 1 is not.

As can be seen, the network allows the two pieces to pass from state 1 to 2 simultaneously. However, since 2 and 3 are states for which precedence is constrained, the assembly of the second piece can only occur when the first piece has reached state 4 at $T6$. The arc $((S2, T4), (S3, T6))$ is incompatible with the arc $((S2, T3), (S3, T5))$ being used by the first piece. Similarly, the arc $((S2, T5), (S3, T7))$ is incompatible with the arc $((S3, T5), (S4, T6))$ being used to complete the assembly of the first piece. The second piece is therefore stored at state 2 for 3 time intervals until it can begin the next assembly step.

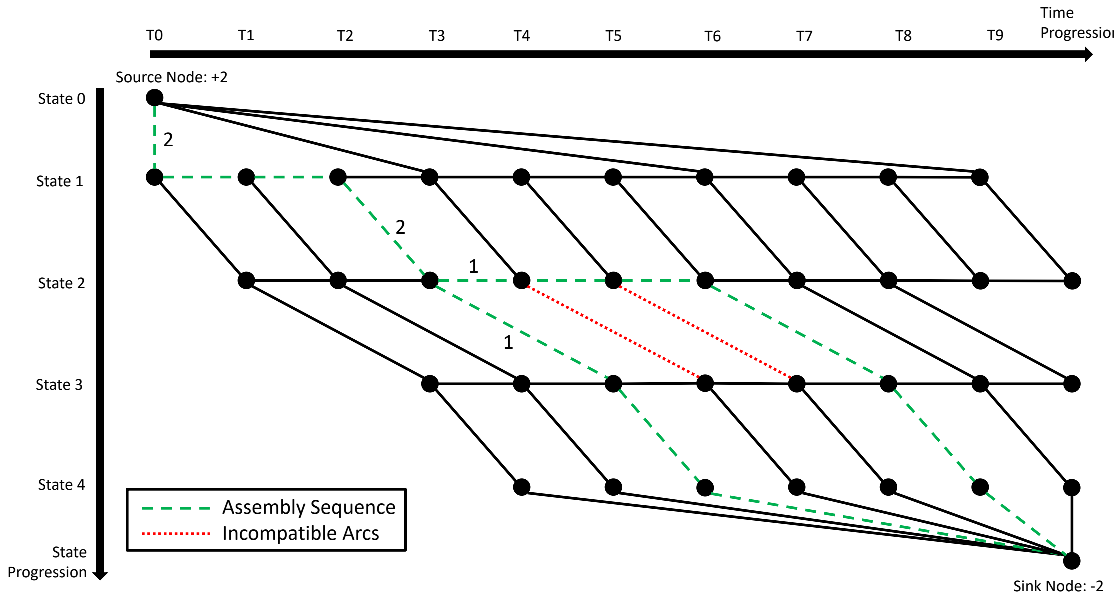

Figure 3.6: Example assembly sequence showing precedence constraints in effect

3.3.4 Task Capacities Across Different Structural Sections

Since structural sections are modelled by individual networks, space systems divided into multiple structural sections are modelled by combinations of networks. Additional constraints must be created to relate these different structural sections together. In particular, task capacity constraints modeling the ability to perform the same task simultaneously can only be implemented across different structural sections, since it is assumed that each individual network models a structural section where each piece must be assembled sequentially. Because pieces can only be assembled one by one within a structural section, simultaneous assembly is only relevant across different structural sections.

The precedence constraints within a structural section defined by Equation 3.3 are related to these new task capacity constraints. The precedence constraint detailed earlier states that only a single piece may be assembled at a time within a structural section, enforcing a 1-by-1 assembly. Therefore, the task capacity constraint (Equation 3.4) simply states that in addition, the number of pieces ongoing a certain assembly task at a certain time across all structural sections SS must be limited by the task capacity of that specific task, K_s, an integer value. T_{st} is the set of arcs related by task capacity constraints at state s and time t. Figure 3.7 displays the arcs related by this constraint for the task $(S2 \rightarrow S3)$ at time $T3$.

$$\sum_{k=1}^{SS} \sum_{(i,j)\epsilon T_{st}} x_{kij} \leq K_s, \ \forall t, \forall s \tag{3.4}$$
$$\text{where } T_{st} = \{(i,j)\epsilon E : (T(i) \leq t, T(j) \geq t, S(i) = s, S(j) \neq s\}$$

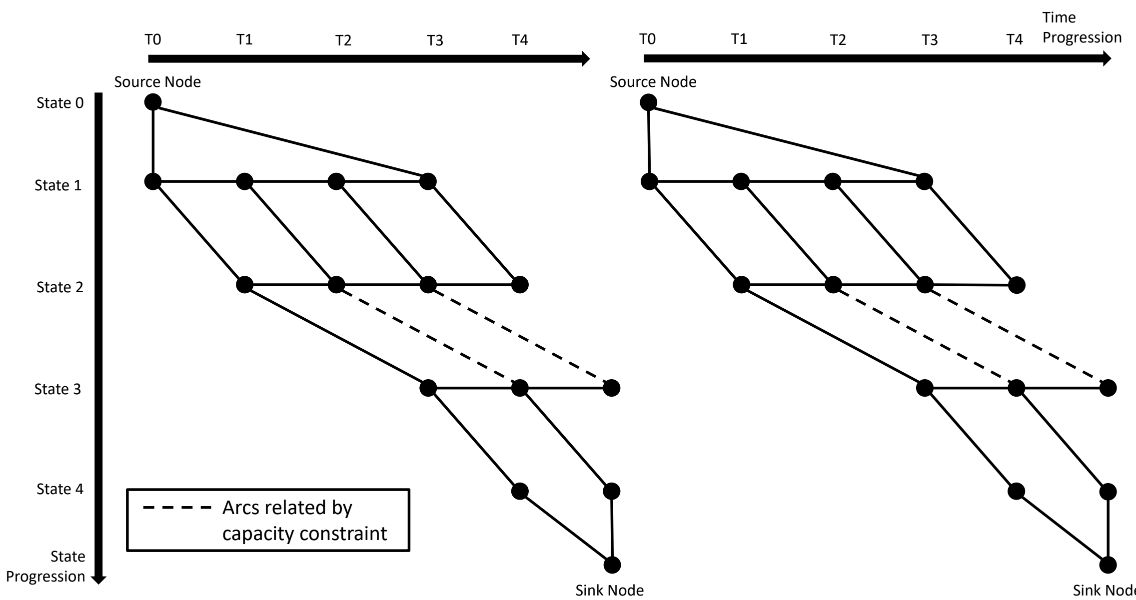

Figure 3.7: ISA network for two structural sections showing arcs related by capacity constraints at t=3.

An example of the assembly process across two different structural sections is shown in Figure 3.8. In this example, all task capacities are set to a limit of 1, such that only one piece may be assembled at a time across both structural sections. Because the assembly of a piece in the first section blocks the arcs of the second section, the piece must wait until $T1$ and $T4$ to begin its assembly tasks. If the capacity were set to 2, both pieces could be assembled simultaneously, since they belong to different sections and are, in this case, not related by precedence constraints across different structural sections.

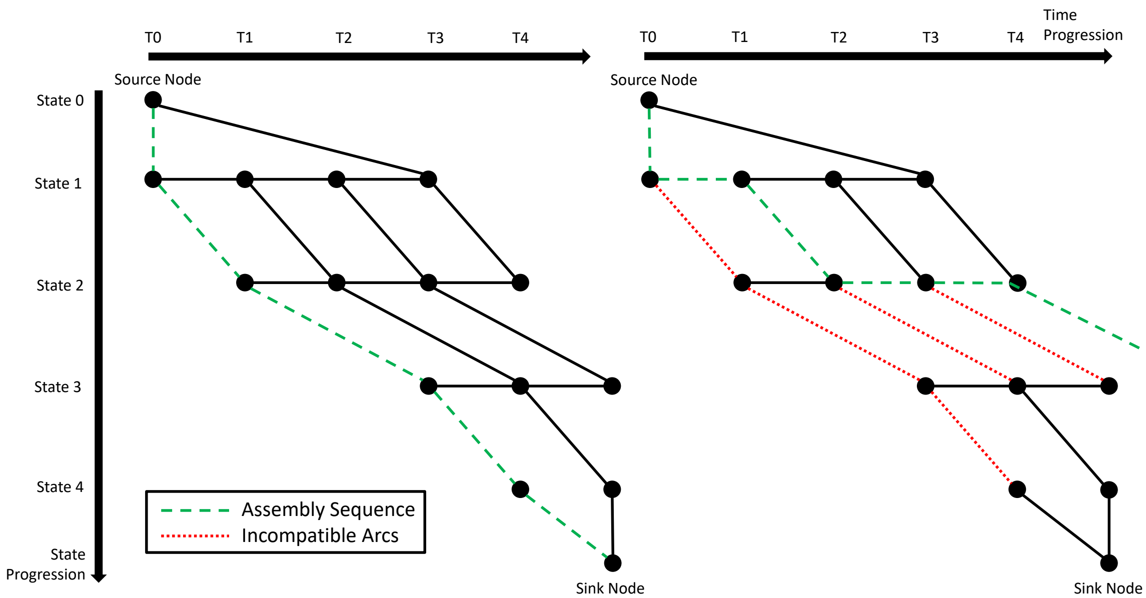

Figure 3.8: Assembly sequence across different structural sections showing task capacity constraint

3.4 Modeling Structural Sections with a Variable Number of Pieces

3.4.1 Piece Merging and Other Issues

Important ISA trades involve the division and delivery of structural sections that are best modeled as a total dimension rather than a fixed number of pieces. Therefore, an effective model of ISA logistics must be able to optimize the division of such a structure into different pieces of variable sizes. The network formulation developed for structural sections with a fixed number of pieces fails to properly model this problem. Replacing the number of pieces per section in the source and sink nodes with a total dimension that must be assembled generates a set of issues that provide erroneous results. A different formulation is therefore needed for these types of structural sections.

The first issue arises from the "merging" of pieces, as shown in Figure 3.9. In this example, the network formulation described earlier is applied to a variable piece division structural section, and the number of pieces replaced with the total dimension to reach. Two delivery types exist, with different capacities. Because the storage and task arcs of the network must be able to accommodate the larger pieces (up to size 8) deliverable by delivery type B, the two pieces of size 4 are allowed merge into a single piece in the arc $((S1, T0), (S2, T2))$. This merging can occur if for example, arc $((S1, T0), (S2, T1))$ is otherwise limited by task capacity constraints across sections. Although this merging conserves the total dimension that must travel from the source to the sink, it has the effect of eliminating the required assembly tasks for the two distinct pieces.

A second issue is caused by arc variables now representing piece size rather than piece count. Task capacity constraints, representing how many pieces can be assembled at the same time, cannot be properly enforced. Capacities are defined in terms of the number of pieces, not in terms of a dimensional quantity. A system able to perform a single task at a time is not able to perform it simultaneously on two pieces because they happen to be half the standard size. Therefore, the model used for fixed division structural sections must be modified.

A partial solution to these problems involves the use of binary variables. An additional network of binary variables can be constructed and linked to the network of integer/piece size variables, such that each "piece size" arc can only take a value greater than zero if the corresponding binary arc is equal to 1. This formulation allows task capacity constraints to be properly applied to the binary variables, by constraining their sum, and not the sum of the size of the pieces, to stay below the task capacity value. However, as shown in Figure 3.10, this formulation does not solve the issues of piece merging. To eliminate merging issues, a final modification to the network is required.

3.4.2 Split Network Formulation

A more complete solution to these issues requires a different network formulation. In this new formulation, nodes are first added to the network such that a node now exists

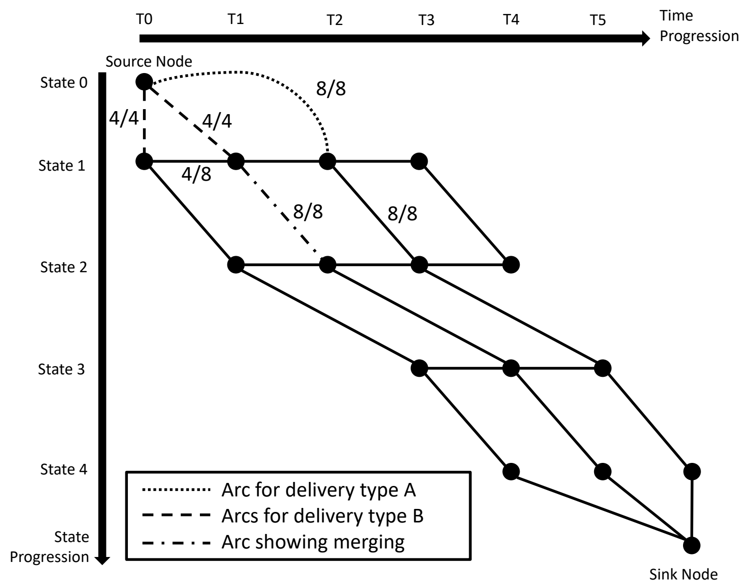

Figure 3.9: Merging of pieces when using basic network to model piece size.

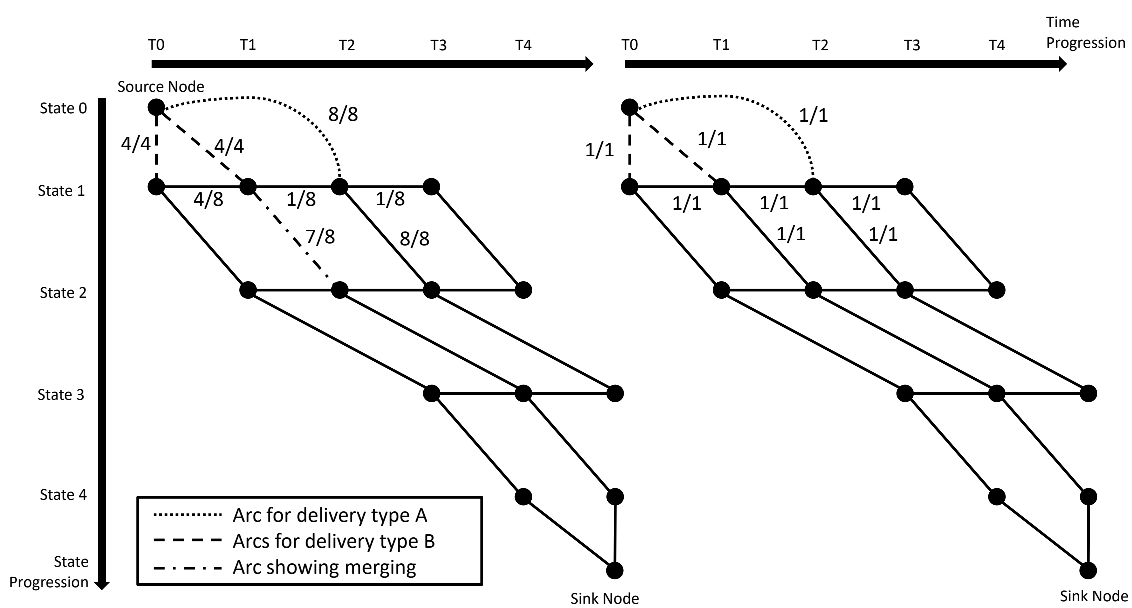

Figure 3.10: Merging of pieces when using a binary network linked to the regular network.

for each state and each time interval, and also for each delivery type. Because different delivery types have different frequencies and capacities, delivery arcs now connect only the source and the network nodes corresponding to each delivery type, forming a "split" network joined at the source and sink, as shown in Figure 3.11. The network is effectively divided into branches corresponding to each delivery type to avoid issues of pieces merging together present in the original formulation.

To avoid merging and properly account for task capacities, additional integer variables b_{ij} are created for each arc in the network and linked to the piece size variables y_{ij} via the capacity constraints of the arcs (Equation 3.5). These variables are related by the same flow conservation rules between themselves, and form therefore a separate but linked network, corresponding to the number of pieces rather than their size, as shown in Figure 3.12. Because the number of pieces is unknown, no flow constraint is created for the source and sink nodes.

$$y_{ij} \leq K_{ij}b_{ij},\ \forall y_{ij} \epsilon E \tag{3.5}$$

By setting the arc constraints on each part of the network according to the delivery capacities of each delivery type, merging no longer occurs unless pieces are delivered in a size lower than the maximum available. Since the objective function of the linear program seeks to minimize the number of deliveries used and the total assembly time, any optimized solution will always place deliveries of a size smaller than the maximum at the end of a delivery schedule, ensuring than no merging can occur.

3.4.3 Precedence Relationships and Task Capacities

In the same way as for the "fixed piece size" network, the task capacity constraint is formulated by relating the variables tracking the number of pieces across different structural sections (Equation 3.6). The variables tracking the number of pieces, b_{kij}, not their size,

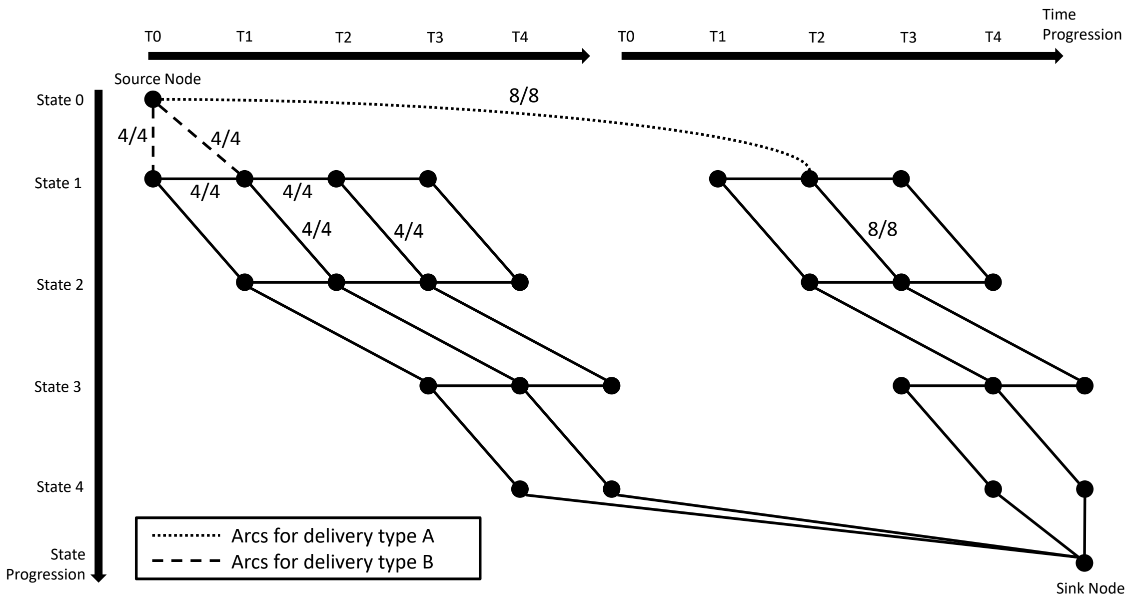

Figure 3.11: Split network for structural sections of ”variable piece size”

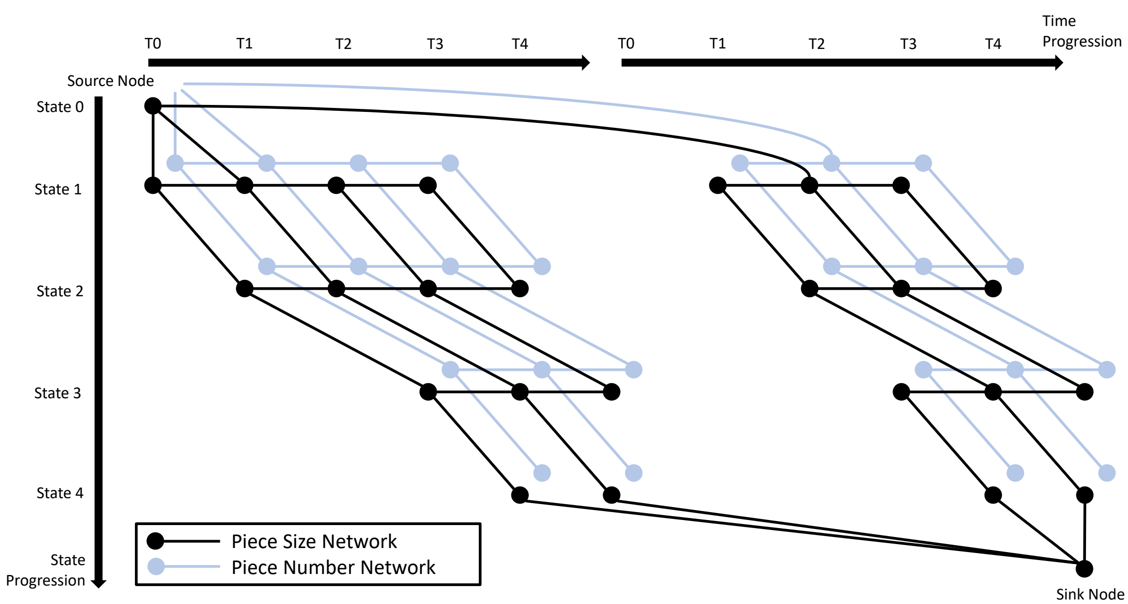

Figure 3.12: Split network showing additional variables tracking the number of pieces.

are used in this equation.

$$\sum_{k=1}^{SS} \sum_{(i,j)\epsilon T_{st}} b_{kij} \leq K_s,\ \forall t, \forall s \qquad (3.6)$$

$$\text{where } T_{st} = \{(i,j)\epsilon E : (T(i) \leq t, T(j) > t, S(i) = s, S(j) \neq s)\}$$

The precedence constraints within a single structural section of "variable piece size" are also enforced similarly to those of the "fixed piece size" networks. However, the piece count variables b_{ij} are used once again to create the precedence constraint. The variables linked in the time cut are set to be exclusive, by limiting their values and sum to zero, enforcing a 1-by-1 precedence relationship (Equation 3.7).

$$b_{ij} \leq 1,\ \forall(i,j)\epsilon P_t, \forall t$$

$$\sum_{(i,j)\epsilon P_t} b_{ij} \leq 1,\ \forall t \qquad (3.7)$$

$$\text{where } P_t = \{(i,j)\epsilon E : (T(i) \leq t, T(j) > t, S(i)\epsilon PST, S(j) \neq min(PST))\}$$

3.5 Modeling Systems as Combinations of Different Structural Sections

3.5.1 Networks Combinations

Figure 3.13 summarizes the two types of networks formulated in previous sections. The first network formulation, used for "fixed piece division" structural sections, tracks when the pieces are assembled, and allows for deliveries to different states, or delivery configurations. The second network formulation tracks both the number of pieces and their sizes, and uses a split formulation, with different branches for different delivery types, and additional variables. Although complex, these networks still retain the natural efficiency of network flow formulations. With these networks formulated, space systems can now be modeled as combinations of them.

Different space systems can be modeled using combinations of sections of "fixed piece

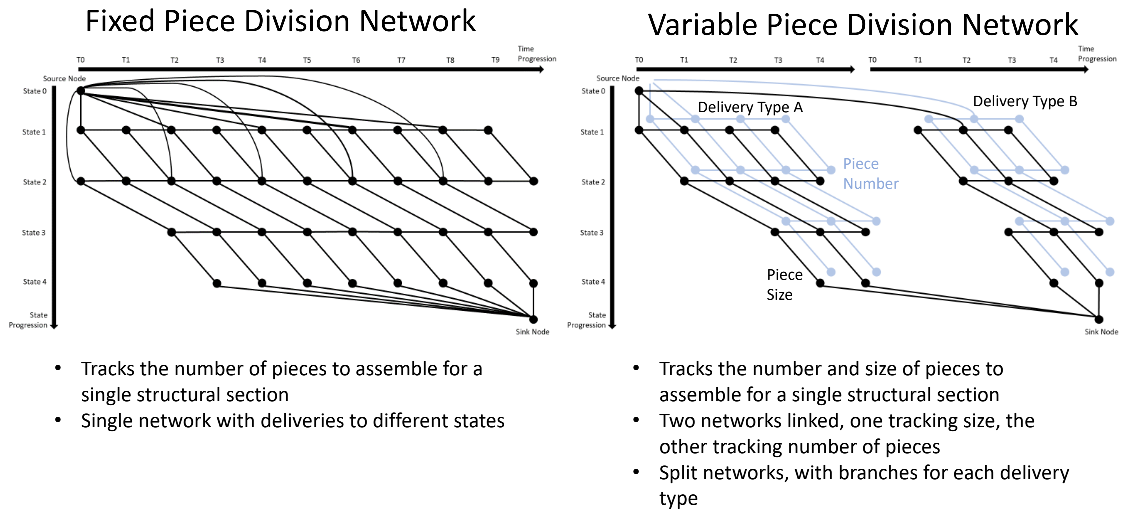

Figure 3.13: Overview of network types used for different structural sections

division" and "variable piece division". The combination of these individual networks form the complete logistics model. These networks can be related by precedence and capacity constraints, indicating the need to complete certain structural sections before others, and the ability to perform assembly tasks simultaneously across different sections. Examples of network arrangements for different types of space systems are shown in Figure 3.14. For example, a space station can be conceptualized as different truss sections of fixed piece number and habitat sections of variable piece size. A surface colony can be divided into pairs of sections for each structure, consisting for example of connector walkways and fixed sized domes. A telescope can consist of a backplane truss and multiple mirrors of variable piece size. The combinations of structural sections can be more or less complex depending on the desired fidelity of the model.

3.5.2 Constraints Between Different Sections

As mentioned previously, task capacity constraints only make sense between different structural sections. However, it may be reasonable to have different or multiple task ca-

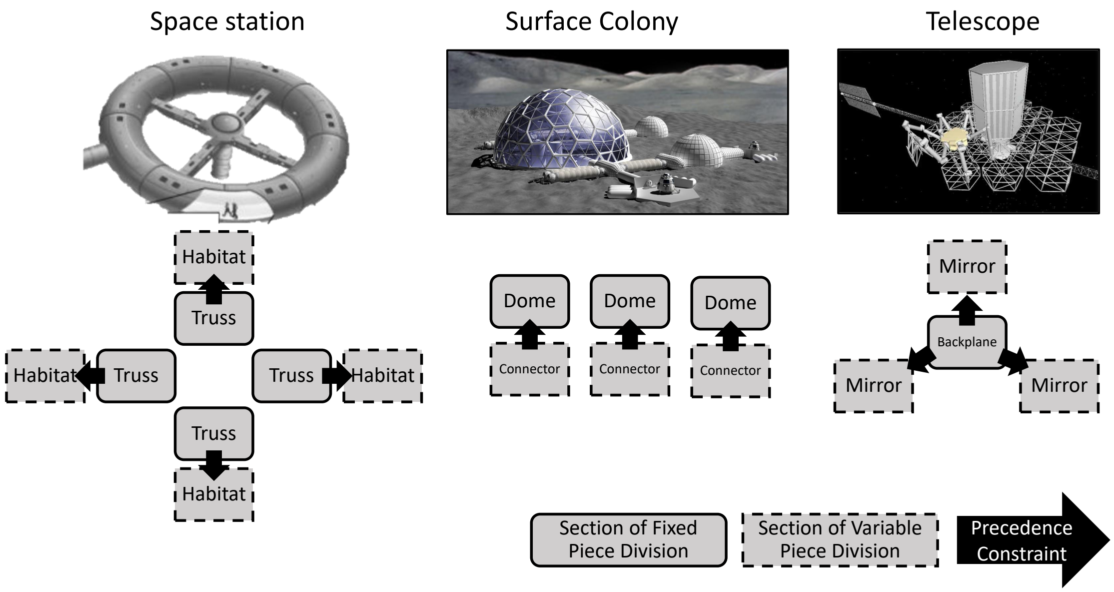

Figure 3.14: Examples of space systems modeled as combinations of structural sections related by precedence constraints

pacities across different groups of structural sections. For example, the robotic arms used for the assembly of a space station may be different for the truss sections and the habitat sections. Other constraints can also be added to the model to account for other relationships between sections. Storage capacities, the constraints on the arcs between two consecutive time intervals at the same state, can be related or not across different structural sections if the storage capacity is considered to be common or independent.

Precedence constraints between different structural sections are the most important relationships. These model the fact that some sections consisting of entirely different pieces may require the completion of another structural section. These relationships are modeled with the addition of a binary variable z_{kt} tracking the completion of a structural section k at a time t, by means of relating it to the sum of the arcs to the sink (the set F_{kt}) and the total number of pieces (or dimension, in the case of ”variable division” sections) N. This variable is then related to both the sum of the sink arcs of the precedent section, and to each assembly arc of the following section (Equation 3.8). In this way, the pieces of the following section cannot begin assembly until all the pieces of the precedent section have been assembled.

$$
\begin{aligned}
& z_{kt+1} \geq z_{kt}, \ \forall k \forall t \\
& z_{kt} \leq \sum_{(i,j) \epsilon F_{kt}} \frac{x_{ij}}{N}, \ \forall k \forall t \\
& z_{kt} \geq b_{kij}, \ \forall k \forall t \forall (i,j) \epsilon P_{kt} \\
& \text{where } F_{kt} = \{(i,j)\epsilon E_k : (T(i) \leq t, T(j) > t, S(j) = max(S)\} \\
& \text{and } P_{kt} = \{(i,j)\epsilon E_k : (T(i) \leq t, T(j) > t, S(i)\epsilon PST, S(j) \neq min(PST))\}
\end{aligned}
\tag{3.8}
$$

3.5.3 Minimizing Assembly Time and Delivery Costs

With a complete network combination model assembled for the different sections of a space system, the objective function can be setup for different objectives. For the fastest assembly time of each piece, weights can be assigned to sink arcs. These weights are proportional to the time at which the arcs occur, such that the optimization will favor completing the assembly as soon as possible. To guarantee that faster completion is preferred, this weights can be quadratic, such that the cost of each sink arc is proportional to the square of the time at which it completes. A time cost factor can multiply this value to adjust the objective function under different scenarios.

The delivery costs can be minimized by assigning weights to additional binary variables d_t tracking the use of each delivery arc. These variables are necessary to maximize the usage of each delivery, so that spreading the same amount of pieces or dimensions across different deliveries incurs a penalty in the objective function as it would in real life. In this way, the solution tries to maximize the usage of each delivery. Linking these new variables is done using Equation 3.9.

$$\sum_{k=1}^{SS} \sum_{(i,j)\epsilon D_{kt}} \frac{x_{kij}}{K_{kij}} \leq d_t, \forall t$$
$$\text{where } D_{kt} = \{(i,j)\epsilon E : (T(i) = 0, S(i) = 0, T(j) = t, S(j)\epsilon DST)\} \tag{3.9}$$

3.6 Complete Model Mathematical Formulation

3.6.1 Symbols Used

Network Notation

k structural section network

(i, j) arc between nodes i and j, can also be written as $((S(i), T(i)), (S(j), T(j)))$ using node State and Time operators

Parameters

c_{kij} Cost of arc variable for network k between nodes i and j

c_t Cost of the delivery occuring at time t

I_{ki} Inflow for node i in network k

K_{kij} Generic capacity of arc between nodes i and j in network k

SCf_s Storage capacity of fixed division networks for state s

SCv_s Storage capacity of variable division networks for state s

L_k Minimum size of piece in variable division structural section k

K_s Capacity of task s, indicating ability to perform it simultaneously

PST_k The set of Precedence STates for a section k for which precedence constraints must be enforced

SST The set of Simultaneity STates for which task capacity constraints must be enforced, common to all structural sections

Sets

E The set of all arcs across all structural sections

E_k The set of all arcs in the structural section network k

V_k The set of all nodes in the structural section network k

D_t The set of delivery arcs occuring at time t and representing a single physical event. $D_t = \{(i,j)\epsilon E : (T(i) = 0, S(i) = 0, T(j) = t, S(j)\epsilon DST)\}$

P_{kt} The set of arcs related by precedence constraints within a section k at time t. $P_{kt} = \{(i,j)\epsilon E_k : (T(i) \leq t, T(j) > t, S(i)\epsilon PST_k, S(j) \neq min(PST_k))\}$

F_{kt} The set of arcs to the sink that are used to track the completion of a structural section k at time t. $F_{kt} = \{(i,j)\epsilon E_k : (T(i) \leq t, S(j) = max(S)\}$

T_{kst} The set of arcs related by task capacity constraints at state s and time t for structural section k. $T_{kst} = \{(i,j)\epsilon E_k : (T(i) \leq t, T(j) \geq t, S(i) = s, S(j) \neq s)\}$

S_{kst} The set of arcs related by storage capacity constraints at state s and time t for structural section k. $S_{kst} = \{(i,j)\epsilon E_k : (S(i) = s, S(j) = s, T(i) = t)\}$

SS_f The set of k indices corresponding to fixed division structural sections

SS_v The set of k indices corresponding to variable division structural sections

Linear Programming Variables

x_{kij} Integer variable tracking the number of pieces in the arc (i,j) in the fixed division network k

y_{kij} Integer variable tracking the size of pieces in the arc (i,j) in the variable division network k

b_{kij} Integer variable tracking the number of pieces in the arc (i,j) in the variable division network k

d_t Binary variable tracking the use of the delivery occurring at time t

z_{kt} Binary variable tracking the completion of section k at time t

3.6.2 Linear Program

- Objective function

$$\text{Minimize} \sum_{(k)\epsilon SS_f} \sum_{(i,j)\epsilon E_k} c_{kij}x_{kij} + \sum_{(k)\epsilon SS_v} \sum_{(i,j)\epsilon E_k} c_{kij}y_{kij} + \sum d_t c_t \tag{3.10}$$

- Flow Conservation Constraints:

$$\sum_{j:(j,i)\epsilon E_k} x_{kji} + I_{ki} = \sum_{j:(i,j)\epsilon E_k} x_{kij},\ \forall i\epsilon V_k, \forall k\epsilon SS_f \tag{3.11}$$

$$\sum_{j:(j,i)\epsilon E_k} y_{kji} + I_{ki} = \sum_{j:(i,j)\epsilon E_k} y_{kij},\ \forall i\epsilon V_k, \forall k\epsilon SS_v \tag{3.12}$$

$$\sum_{j:(j,i)\epsilon E_k} b_{kji} = \sum_{j:(i,j)\epsilon E_k} b_{kij},\ \forall i\epsilon V_k, \forall k\epsilon SS_v \tag{3.13}$$

- Link between piece size and piece number variables for variable division networks:

$$L_k b_{kij} \leq y_{kij} \leq K_{kij} b_{kij},\ \forall (i,j)\epsilon E_k, \forall k\epsilon SS_v \tag{3.14}$$

- Precedence within Structural Section Networks:

$$x_{kij} \leq 1,\ \forall (i,j)\epsilon P_{kt}, \forall t, \forall k\epsilon SS_f \tag{3.15}$$

$$\sum_{(i,j)\epsilon P_{kt}} x_{kij} \leq 1,\ \forall t, \forall k\epsilon SS_f \tag{3.16}$$

$$b_{kij} \leq 1,\ \forall (i,j)\epsilon P_{kt}, \forall t, \forall k\epsilon SS_v \tag{3.17}$$

$$\sum_{(i,j)\epsilon P_{kt}} b_{kij} \leq 1,\ \forall t, \forall k\epsilon SS_v \tag{3.18}$$

- Precedence across Structural Section Networks

$$z_{kt+1} \geq z_{kt},\ \forall k \forall t \tag{3.19}$$

$$z_{kt} \leq \sum_{(i,j)\epsilon F_{kt}} \frac{x_{ij}}{N},\ \forall k \forall t \tag{3.20}$$

$$b_{kij} \leq z_{kt},\ \forall k \forall t \forall (i,j)\epsilon P_{kt} \tag{3.21}$$

- Storage Capacity Across Structural Sections of the same type

$$\sum_{k\epsilon SS_f} \sum_{(i,j)\epsilon S_{kst}} x_{kij} \leq SCf_s,\ \forall t, \forall s \tag{3.22}$$

$$\sum_{k\epsilon SS_v} \sum_{(i,j)\epsilon S_{kst}} b_{kij} \leq SCv_s,\ \forall t, \forall s \tag{3.23}$$

- Task Capacity Across Structural Sections of the same type

$$\sum_{k\epsilon SS_f} \sum_{(i,j)\epsilon T_{kst}} x_{kij} \leq TCf_s,\ \forall t, \forall s\epsilon SST \tag{3.24}$$

$$\sum_{k\epsilon SS_v} \sum_{(i,j)\epsilon T_{kst}} b_{kij} \leq TCv_s,\ \forall t, \forall s\epsilon SST \tag{3.25}$$

- Common Task Capacity Across All Structural Section Networks

$$\sum_{k\epsilon SS_f} \sum_{(i,j)\epsilon T_{kst}} x_{kij} + \sum_{k\epsilon SS_v} \sum_{(i,j)\epsilon T_{kst}} b_{kij} \leq CTC_s,\ \forall t, \forall s\epsilon SST \tag{3.26}$$

- Delivery Capacity Across Across All Structural Section Networks

$$\sum_{k\epsilon SS_f} \sum_{(i,j)\epsilon D_t} \frac{x_{kij}}{K_{kij}} + \sum_{k\epsilon SS_v} \sum_{(i,j)\epsilon D_t} \frac{y_{kij}}{K_{kij}} \leq d_t \tag{3.27}$$

CHAPTER 4

SAMPLE PROBLEM - ARTIFICIAL GRAVITY SPACE STATION

4.1 Rationale for Choice of Sample Problem

The methodology described in Chapter 3 was applied to the problem of assembling an Artificial Gravity Space Station (AGSS). Such systems have been a subject of interest to space exploration enthusiasts and professionals for some time, and are ubiquitous in works of science-fiction. The reason for their popularity lies in the simplicity of the underlying physics: the station or part of it is rotated such that the centripetal acceleration experienced by its occupants partially simulates the effects of gravity. If it were to be built on Earth, an AGSS would constitute a relatively trivial construction project. However, due to the size and scale required to achieve meaningful artificial gravity, and the limitations of launch vehicles, an AGSS must be assembled in space.

Despite its size and complexity, such a project would not be an unrealistic endeavour in the mid-term future. In the context of a planned long-term human presence on the surface of the Moon, such a station could provide mission support in the form of an emergency escape location, and a waypoint between the surface and the Earth. Assuming a continuous human presence, micro-gravity mitigation would be a critical requirement for the crew's health and well being. In addition, the ISS has proven that ambitious, multi-year ISA projects can succeed. With advances in launch vehicle re-usability increasing the delivery capability to orbit, and new technologies allowing the implementation of more complex ISA strategies, assembling a large AGSS in orbit will soon no longer be restricted to the realm of science-fiction.

Because assembling such a large system would take years and make use of advanced ISA strategies, a number of design trades will need to be considered well in advance of its construction. Although some capabilities might exist independently of the AGSS project, it is reasonable to assume that an ISA infrastructure will need to be sized, and launch schedules planned ahead. The impact of various parameters on the assembly time and delivery costs need to be determined in order to make decisions on the capabilities needed. The application of the methodology described in Chapter 3 to this problem offers answers to those questions and others.

4.2 Conceptual Space Station Sizing

A conceptual design of an AGSS was performed as part of a previous class project. The design follows a spinning wheel architecture, where a habitat ring torus is connected to a central core via four truss arms, and the entire station rotates together. The AGSS is sized from its artificial gravity requirements, to achieve an acceleration of 0.34g (similar to the surface of Mars) while staying within a "comfort zone", limited by the gravity gradient (the difference between the acceleration felt at the head and feet) experienced by its occupants [38]. This design is shown in Figure 4.1.

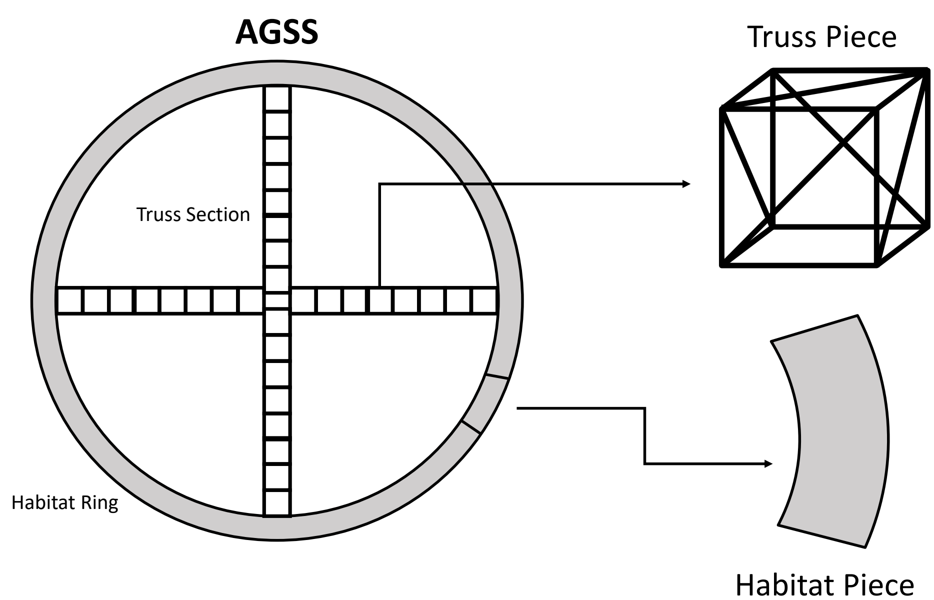

Figure 4.1: Artificial Gravity Space Station Components

The station has a diameter of 45m, and rotates at 3.7rpm in order to generate 0.34g of centripetal acceleration inside the outer ring. The AGSS mainly consists of four truss arms and the habitat ring. Each of the four truss arms is divided into 8 "pieces" formed of struts arranged in a cubic pattern of 2.8m length. The ring consists of a habitat torus of 4.5m diameter. The division of this ring is variable: it can be divided into sections of as low as 2 degrees, for a maximum of 180 pieces.

4.3 Logistics Model Configuration

Using the methodology outlined in Chapter 3, the AGSS is modeled as 8 separate structural sections, each with its own network. Four fixed piece division networks correspond to the four truss arms. The network used for truss sections is shown in Figure 4.2. The four remaining structural sections correspond to the habitat ring. Each truss section is related to a corresponding habitat section by a precedence relationship. The network used for habitat sections is shown in Figure 4.3. The arrangement of these networks and precedence relationships to form the model of the AGSS is shown in Figure 4.4.

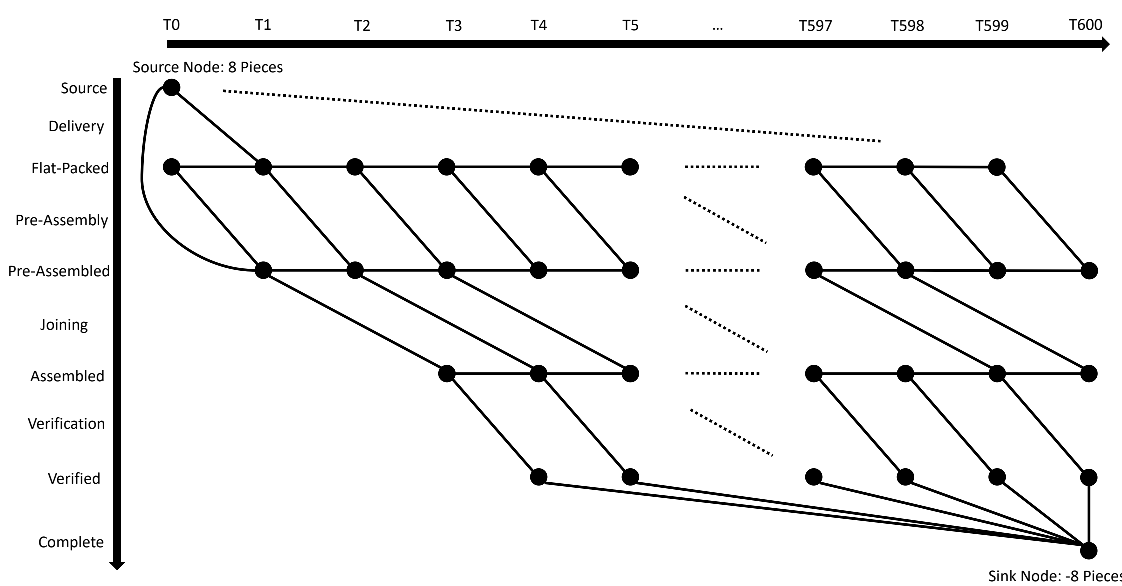

Figure 4.2: Network used for an AGSS truss structural section

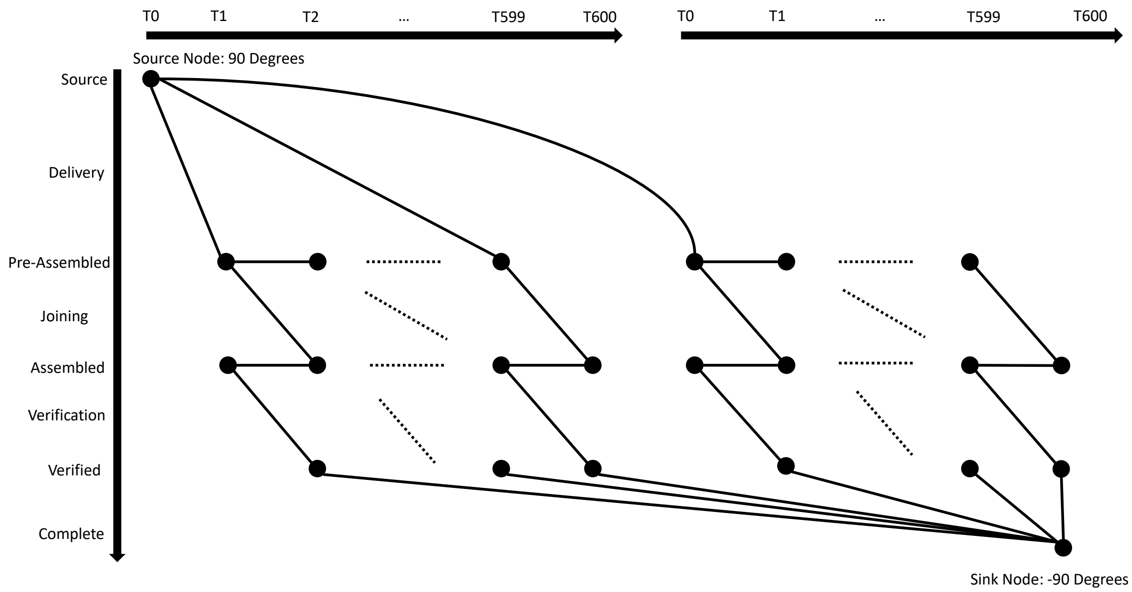

Figure 4.3: Network used for an AGSS Habitat structural section

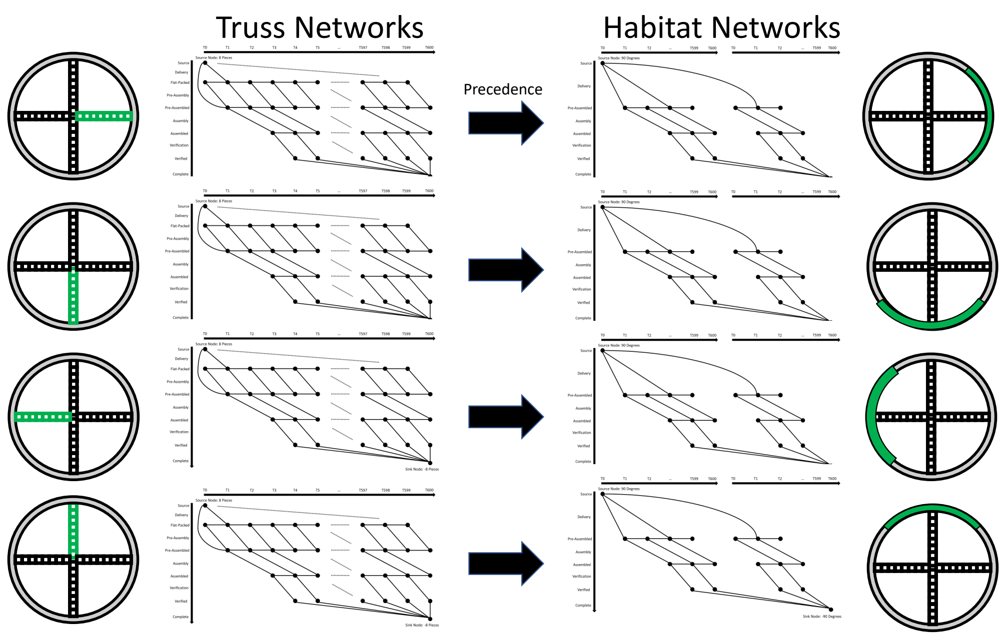

Figure 4.4: Combination of networks and precedence relationships that form the complete AGSS model

Each truss network must assemble eight pieces. As described previously, each piece is interchangeable, and the assembly sequence is linear: pieces cannot be assembled simultaneously within the same truss. Two delivery configurations are available to the truss pieces: flat-packed and pre-assembled. The states of assembly for these sections are: Flat-Packed, Pre-Assembled, Assembled, Verified. The corresponding intermediate tasks are Delivery, Pre-Assembly, Joining, and Verification. Of all the tasks, only pre-assembly does not require precedence: a truss piece may undergo pre-assembly even while another piece is being completed.

The ring is divided into 4 sections for which a total of 90 degrees must be delivered and assembled, using the variable division networks. The minimum size of each piece is 2 degrees, such that the maximum number of pieces for each section is 45. A precedence constraint is implemented between each habitat network and the corresponding truss network, such that habitat pieces can only be assembled after their truss network has been completed. Habitat pieces deliver only to the pre-assembled state. As a result, their available states of assembly are: Pre-Assembled, Assembled, and Verified, and only two tasks are required: Joining, and Verification. As described previously, the network has different branches for each delivery type, as well as an accompanying binary network (not shown) to avoid piece merging issues.

4.4 Scenarios

4.4.1 Baseline Scenario

Baseline Scenario Parameters

A baseline scenario was first defined so that design trades could be conducted relative to this scenario. An overview of the scenario parameters is shown in Table 4.1. Each Time Unit (TU) is set to represent 3 days, to minimize the size of the linear program while maintaining enough resolution in setting the duration of the various tasks. The total time is set to 600

Table 4.1: Parameters of Baseline Scenario for AGSS Assembly

General Parameters	**Value**
Time Unit	3 days
Total Time	600 TU (5.75 years)
Time Cost Factor (Time Critical)	11
Time Cost Factor (Cost Critical)	1
Delivery Capability Parameters	**Value**
Delivery Types	2
Delivery Frequencies	1/5 and 1/30 TU
Delivery Capacity for Flat-Packed Truss Pieces	8 and 16 pieces
Delivery Capacity for Pre-Assembled Truss Pieces	1 and 2 pieces
Delivery Capacity for Hab Sections	4 and 8 degrees
Delivery Costs (Time Critical)	10 and 15
Delivery Costs (Cost Critical)	1000 and 1500
ISA Capability Parameters	**Value**
Flat Packed State	
Truss Sections Storage Capacity	16 pieces
Storage Cost Factor	0.01
Pre-Assembly Task	
Common Capacity	1 piece
Truss Sections Task Duration	3 TU
Truss Sections Task Capacity	1 piece
Pre-Assembled State	
Truss Sections Storage Capacity	4 pieces
Hab Sections Storage Capacity	1 piece
Storage Cost Factor	0.02
Joining Task	
Common Capacity	1 piece
Truss Sections Task Duration	5 TU
Truss Sections Task Capacity	1 piece
Hab Sections Task Duration	3 TU
Hab Sections Task Capacity	1 piece
Assembled State	
Truss Sections Storage Capacity	4 pieces
Hab Sections Storage Capacity	1 piece
Storage Cost Factor	0.03
Verification Task	
Common Capacity	1 piece
Truss Sections Task Duration	2 TU
Truss Sections Task Capacity	1 piece
Hab Sections Task Duration	1 TU
Hab Sections Task Capacity	1 piece

Time Units, or a total 5.75 years. This total time allows the completion of the assembly under every scenario, and keeps the mathematical model manageable while allowing trades of cost versus time to be performed, such as the use of less frequent deliveries of more capacity.

Two delivery types are considered. A first delivery type occurs every 5 Time Units (15 days) and is capable of delivering 8 flat-packed truss pieces, 1 pre-assembled truss piece, and up to 4 degrees of habitat arc length. The second delivery type occurs only every 30 Time Units (90 days). However, it can deliver 16 flat-packed truss pieces, 2 pre-assembled truss pieces, and up to 8 degrees of habitat arc length. These two delivery types are sized to broadly represent a small/medium reusable launch vehicle (such as the SpaceX Falcon9) and a heavy launch vehicle (such as the Falcon Heavy), respectively. The different frequencies and capacities allow a tradeoff between speed and cost effectiveness to be conducted.

For each task, the capacity across different trusses and the capacity across different habitat sections is set to a single piece. In addition, the common capacity of each task across the entire system is set to a single piece as well. As a result, for this baseline scenario, only a single piece may be undergoing each task at a particular time: a habitat section may be assembled while a truss section (of a different arm) is verified, but two pieces of any kind cannot be assembled at the same time. At the flat-packed state, 16 truss pieces can be stored. At every other state, only 4 truss pieces may be stored. Habitat pieces are limited to 1 piece being stored at each state per arm.

In the absence of any existing data, the duration of assembly tasks must be estimated. For the truss sections, the duration of the pre-assembly, assembly, and verification tasks are set to 3 TU (9 days), 5 TU (2 weeks), and 2 TU (6 days), respectively. These are conservatives estimates of the time it would take to secure, move, position, weld, connect, link, verify and test multiple struts to form a truss pattern in orbit. For the habitat section, the assembly task is set to 3 TU (9 days) and the verification task to 1 TU (3 days).

The total time to assemble and the cost of deliveries are important metrics of the effectiveness of the assembly process. Because of the difficulty in reducing the total time of assembly and the cost of deliveries into a common unit, the baseline and subsequent derived scenarios are explored with two different objective functions, one emphasizing time and the other emphasizing delivery costs. Under the time critical condition, assembling each habitat piece as soon as possible is prioritized, in order to achieve a partial functionality while under construction. Under the cost critical condition, the time to assemble is a secondary objective and reducing delivery costs is prioritized.

The value of the time cost factor described in Chapter 3 is set to 11 and to 1 for the time critical and cost critical scenarios, respectively. The delivery costs for each delivery type are set to 10 and to 15 in the time critical condition, and to 1000 and 1500 for the cost critical condition. The two delivery types are weighted such that the higher payload, lower frequency type is more cost effective relative to its delivery capacity. The storage cost factors are kept constant for both scenarios, at 0.01, 0.02, and 0.03 for each state.

Baseline Scenario Results

Results were obtained by solving the linear integer program with the commercial optimization solver *Gurobi* on a Intel Core I7-7700T quad-core processor and a maximum of 16GB of RAM. In some cases, the solution had to be interrupted before reaching maximum optimality due to time constraints. The delivery arrangements and assembly sequences calculated can be displayed in the form of a Gantt chart, as shown in Figure 4.5. Deliveries are indicated by a red star and the corresponding number of pieces delivered. For habitat sections, the size of the piece delivered can be shown instead. Yellow storage bars represent pieces awaiting assembly tasks. The height of the storage bars represents the number of pieces in storage at that particular state and time. Assembly tasks are represented by black bars, with different durations represent by different widths. The completion of the assembly process for a piece is marked by a green star. In Figure 4.5, two tasks can be seen

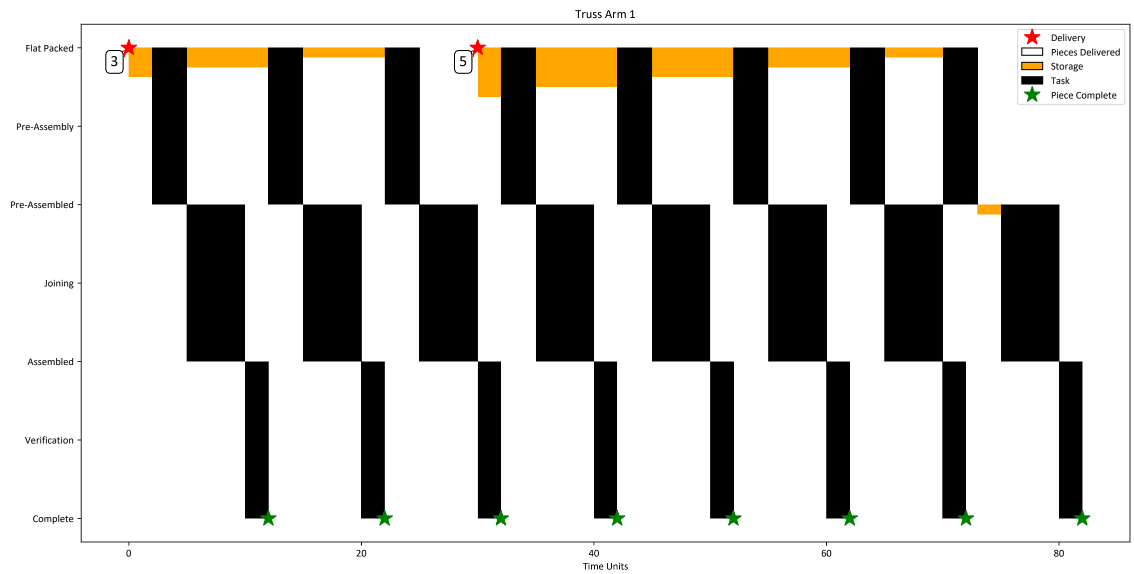

Figure 4.5: Example Gantt chart for truss assembly and delivery schedule

to occur simultaneously around the 70 TU mark: the pre-assembly of a piece occurs while another is still being verified. This assembly schedule is within the constraints of the problem, since the pre-assembly task is not considered to be part of the precedence required for the assembly of a truss piece.

The full results for the baseline under the Time Critical and Cost Critical conditions are shown in Figures 4.6 and 4.7, respectively in the form of an individual Gantt-type chart for each structural section. Under the time critical objective, the assembly takes a total time of 457 TU (3.75 years) and uses a total of 80 deliveries. As can be seen in Figure 4.6, the assembly for truss arms 1 and 4 is completed early, around TU 80 (approximately 8 months in). The other two arms, 2 and 3, have their pieces delivered early and held in storage for a long time, while the habitat sections attached to arms 1 and 4 begin delivery and assembly. This allows the usage of the distinct joining and verification task ”workers” to be maximized, by simultaneously joining a truss piece while verifying a habitat piece attached to a different arm.

Under the Cost Critical condition shown in Figure 4.7, the baseline scenario finishes assembly in 574 TU (4.7 years). The assembly uses a total of 74 deliveries, with 54 of the higher frequency type and 20 of the higher capacity type. Trusses 1 and 3 complete assembly quickly around 80 TU while the other two are spread out longer over time. To minimize costs, the use of higher capacity deliveries for the habitat pieces is maximized over the allowable time period, beginning when the truss sections are completed.

4.4.2 Improved In-Space Assembly Capability Scenarios

To observe the impact of varying the available ISA on the two core metrics of interest, assembly time and delivery costs, results were obtained for additional scenarios derived from the baseline. The ISA capability was first varied by separately doubling task capacities, doubling storage capacities, and reducing task durations, and by combining these three improvements under a single scenario. The detailed parameter inputs for these test cases are shown in Table 4.2.

Double Task Capacity Scenario

Under this scenario, each task capacity, as well as the common task capacity, are doubled, so that two pieces can undergo a same task at the same time. Figure 4.8 displays the results obtained for this scenario under the Time Critical Condition. The AGSS completes assembly after 434 TU (3.5 years) using 80 deliveries. The time gained relative to the baseline is of 20 TU (60 days). Results show two observable trends. First, the faster assembly of the truss arms, which are all completed in approximately 120 TU (1 year), much faster than the baseline, thanks to the ability to assemble two different arms simultaneously. Second, the habitat sections are delivered in smaller pieces on the same delivery events, allowing for the two available worker units for each task to simultaneously assemble pieces for different habitat sections. Despite the additional time required for the individual assembly of these more numerous pieces, there is a net gain in the time taken for the complete assembly process.

Figure 4.9 displays the results for the Double Task Capacity scenario under the Cost

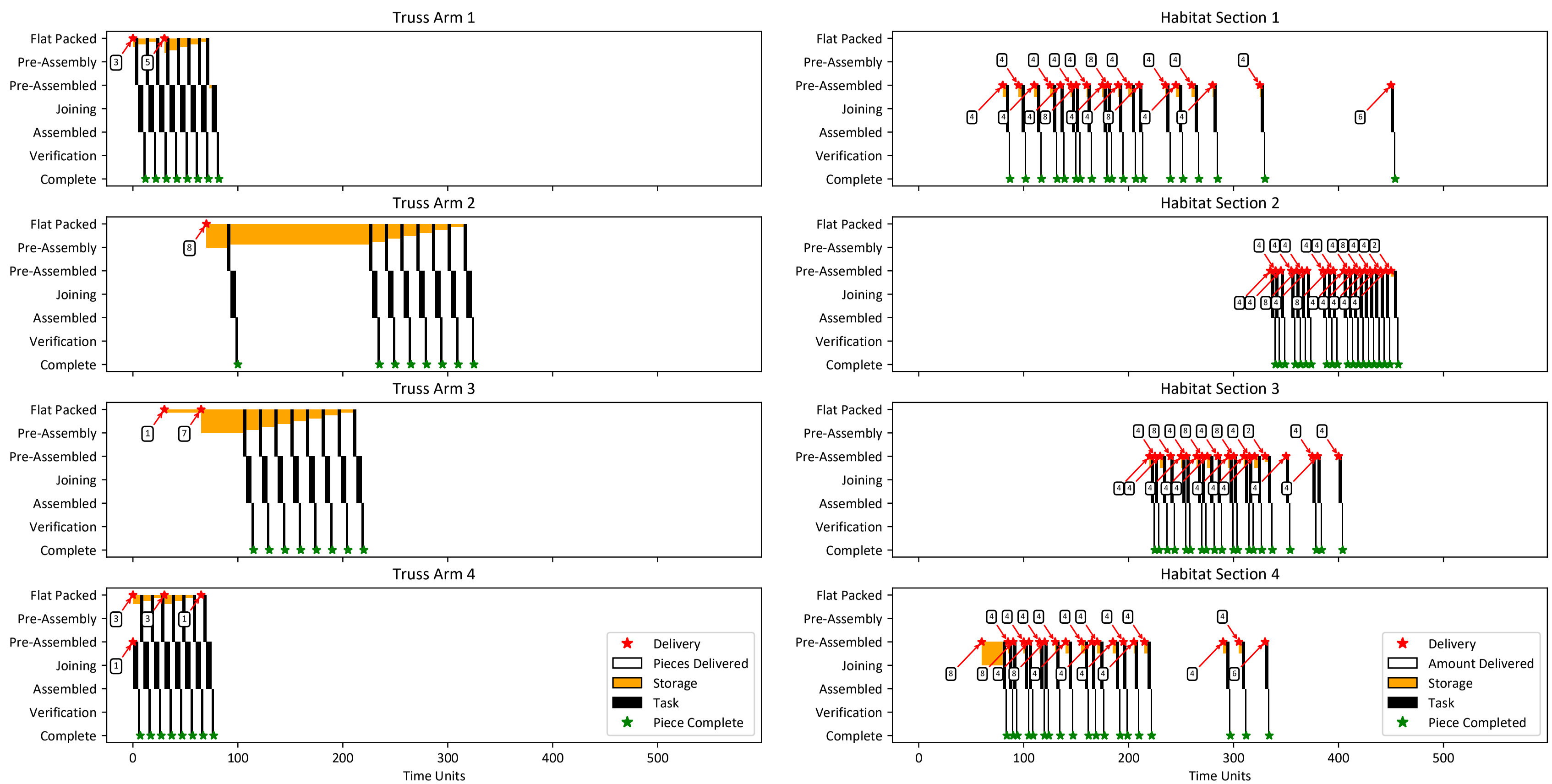

Figure 4.6: Results for the baseline scenario under the time critical condition

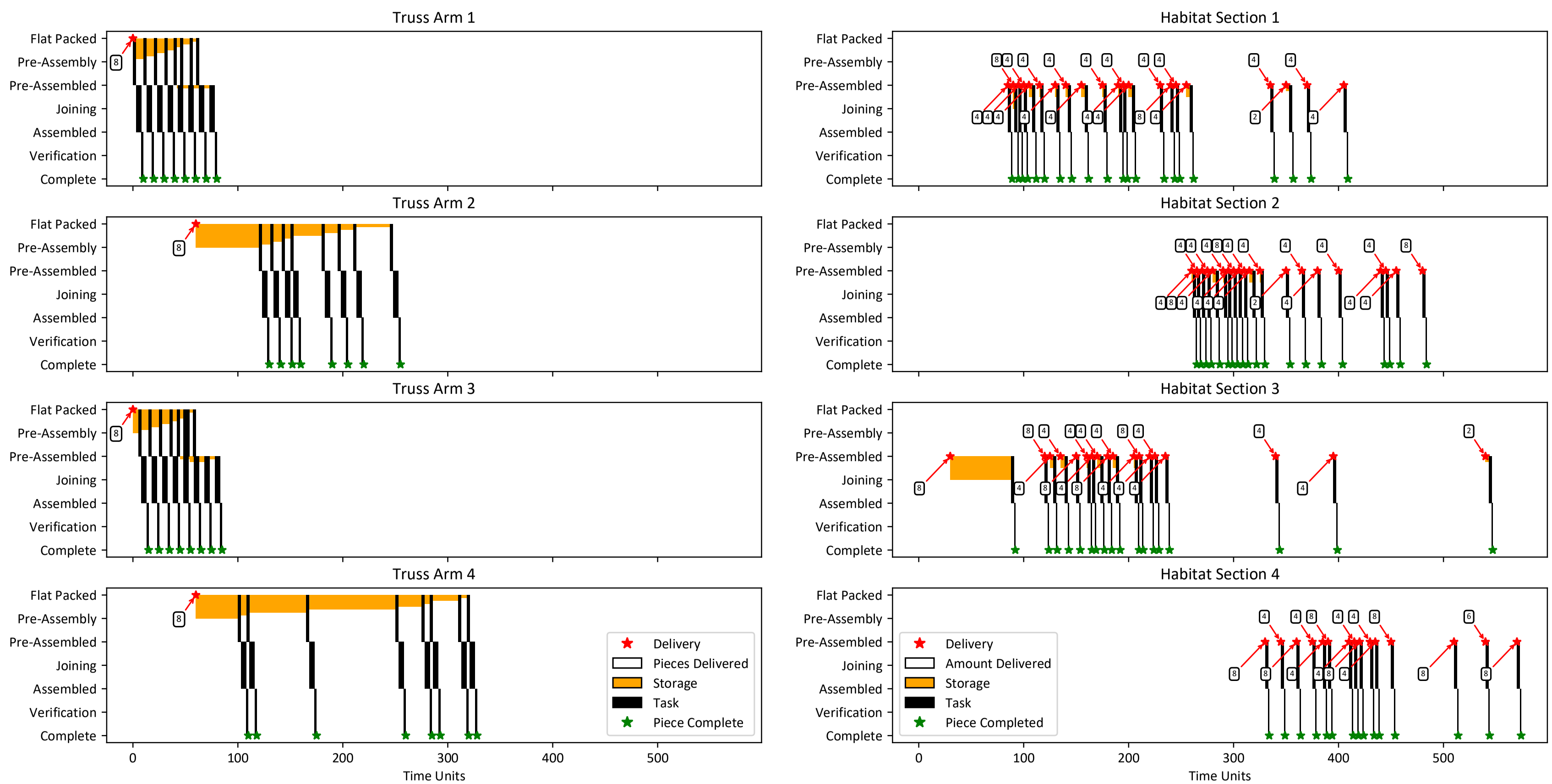

Figure 4.7: Results for the baseline scenario under the cost critical condition

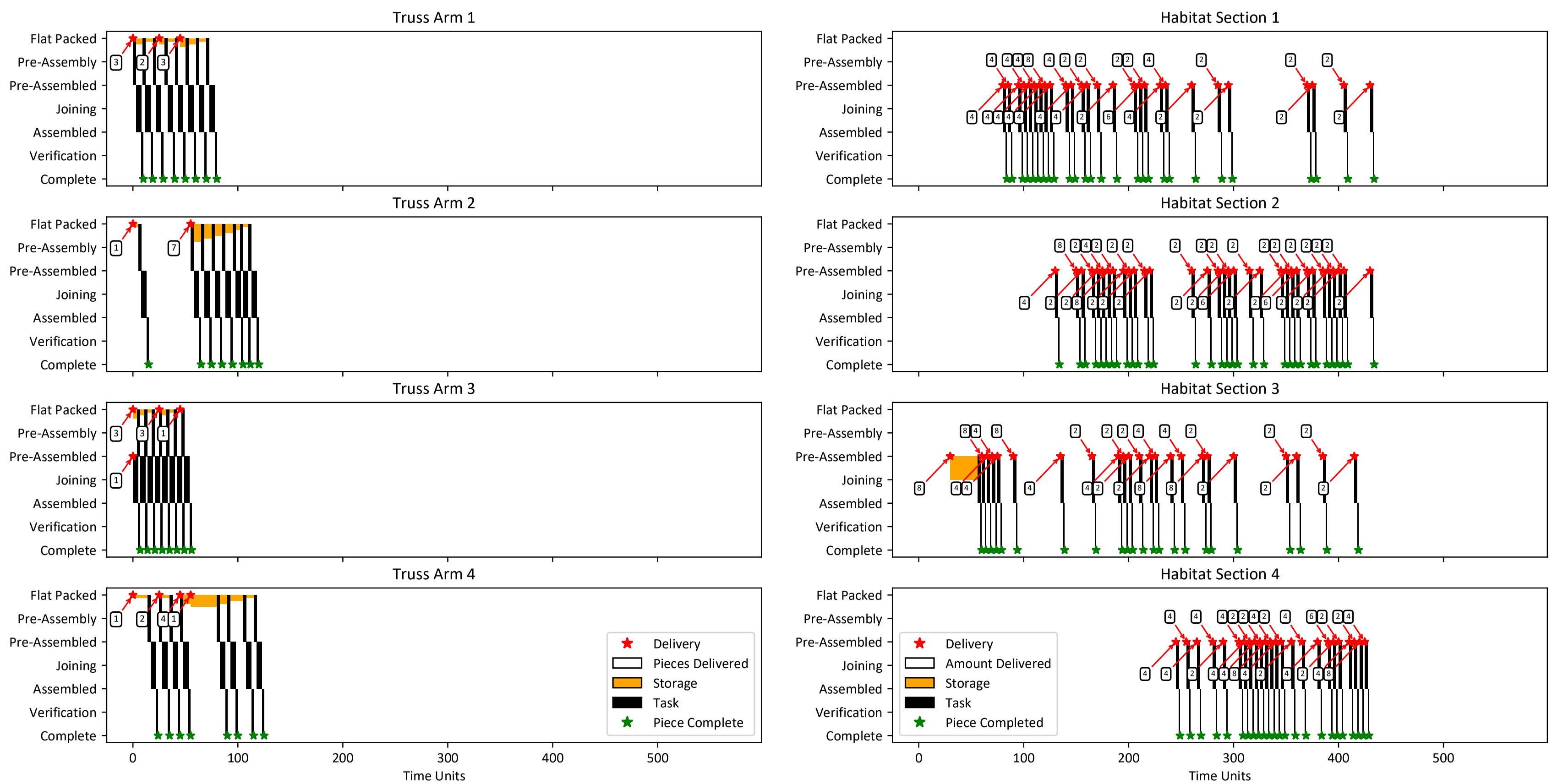

Figure 4.8: Results for the double task capacity scenario under the time critical condition

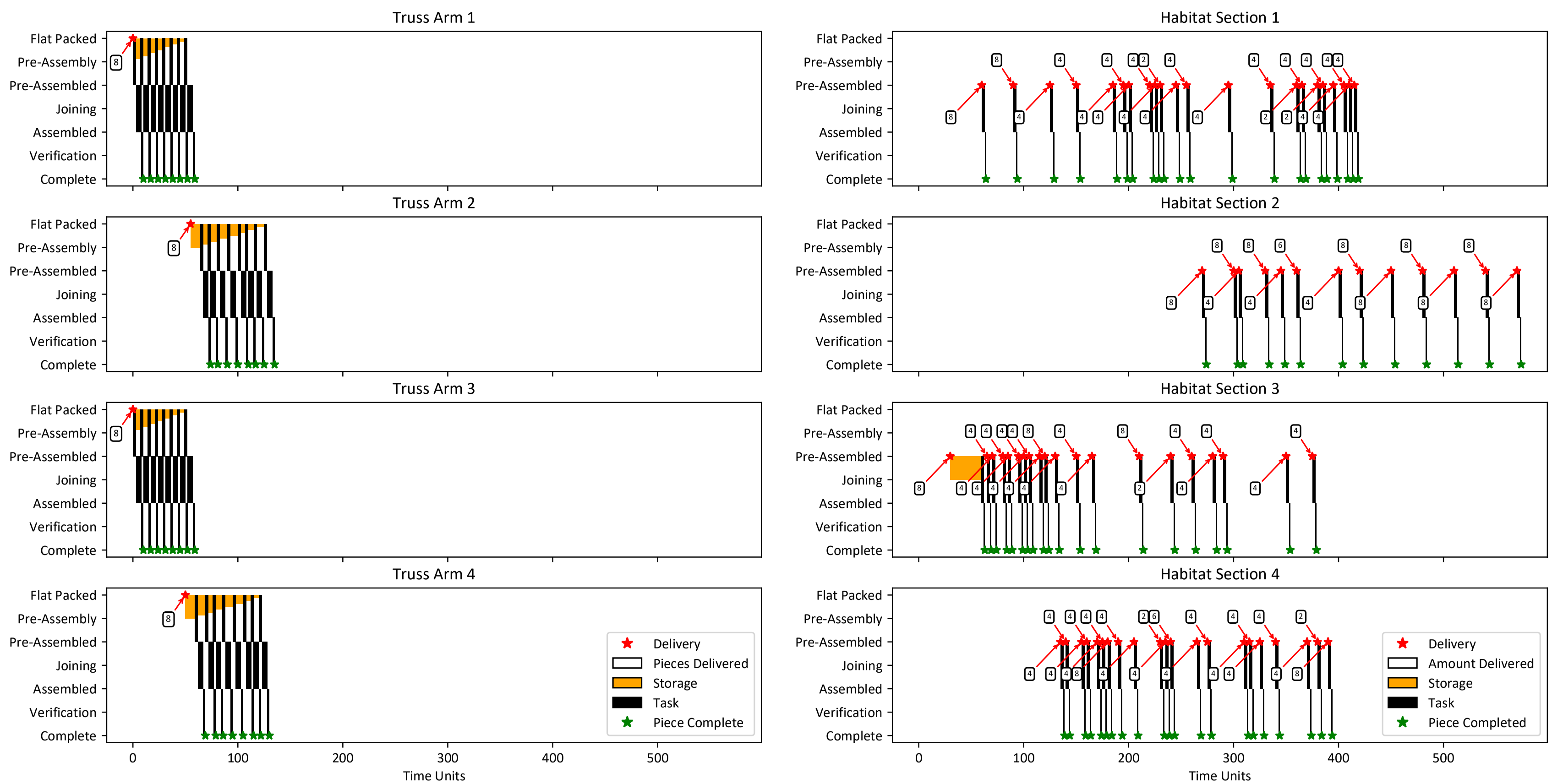

Figure 4.9: Results for the double task capacity scenario under the cost critical condition

Table 4.2: Parameters for Advanced ISA Capability Scenarios

Parameters	2x Capacity	2x Storage	<Duration	**Combined**
Flat Packed State				
Truss Storage Capacity	16 pieces	**32 pieces**	16 pieces	**32 pieces**
Pre-Assembly Task				
Common Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Truss Task Duration	3 TU	3 TU	**2 TU**	**2 TU**
Truss Task Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Pre-Assembled State				
Truss Storage Capacity	4 pieces	**8 pieces**	4 pieces	**8 pieces**
Hab Storage Capacity	1 piece	**2 pieces**	1 piece	**2 pieces**
Assembly Task				
Common Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Truss Task Duration	5 TU	5 TU	**3 TU**	**3 TU**
Truss Task Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Hab Task Duration	3 TU	3 TU	**2 TU**	**2 TU**
Hab Task Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Assembled State				
Truss Storage Capacity	4 pieces	**8 pieces**	4 pieces	**8 pieces**
Hab Storage Capacity	1 piece	**2 piece**	1 piece	**2 piece**
Verification Task				
Common Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Truss Task Duration	2 TU	2 TU	**1 TU**	**1 TU**
Truss Task Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**
Hab Task Duration	1 TU	1 TU	1 TU	1 TU
Hab Task Capacity	**2 pieces**	1 piece	1 piece	**2 pieces**

Critical condition. With the objective of minimizing cost, this scenario finishes in 574 TU (4.7 years) using 74 deliveries, exactly the same values obtained for the baseline scenario under the same conditions. Comparing Figures 4.9 and 4.7 reveals that the scenarios differ in the time to assemble the truss arms, and the amount of storage time used. The doubled task capacity allows trusses to be assembled simultaneously and complete faster, allowing a longer time frame in which to maximize the use of the more efficient higher capacity delivery type. Although this scenario does not improve the total delivery cost, it slightly reduces the storage costs.

Double Storage Scenario

Under the Double Storage scenario, the storage capacities at each state for both truss and habitat pieces are doubled. Figure 4.10 display the results for the Double Storage Capacity scenario under Time Critical conditions. This scenario completes assembly in 457 TU (3.75 years) and uses a total of 80 deliveries, a result remarkable for exactly matching the values of the unmodified baseline scenario. Without an accompanying increase of ISA or delivery capability, increasing the available storage has no effect on these two metrics. However, it may have a significant effect if combined with other improvements.

The results obtained for this scenario under the Cost Critical conditions are shown in Figure 4.11. As for the baseline and the double capacity scenario, the assembly finishes in 574 TU (4.7 years) using 74 deliveries. Although the assembly makes use of the increased storage capacity (notably to hold two large habitat pieces during the assembly of section 2), there is no visible trend differentiating it from the baseline. Once again, increased storage capacities have little effect absent simultaneous improvements in other parameters.

Reduced Task Duration Scenario

In this scenario, the duration of all tasks (with the exception of the habitat verification task) was reduced. Figure 4.12 displays the results obtained under this scenario for the Time Critical condition. The assembly of the AGSS is completed after 418 TU (3.4 years) using 82 deliveries, a gain of 39 TU (4 months) over the baseline conditions. The first truss arm is completed very quickly after 34 TU, allowing the assembly of the corresponding habitat sections to proceed immediately. Because these habitat sections begin assembly early, the assembly of the truss sections after the first is visibly slower: these must ”compete” with habitat sections under the task capacity constraint of a single piece. On the other end, habitat section number 3, which is assembled last, is completed faster than the others, as it is not being assembled concurrently with any other section.

This last ISA capability improvement scenario under Cost Critical conditions offers

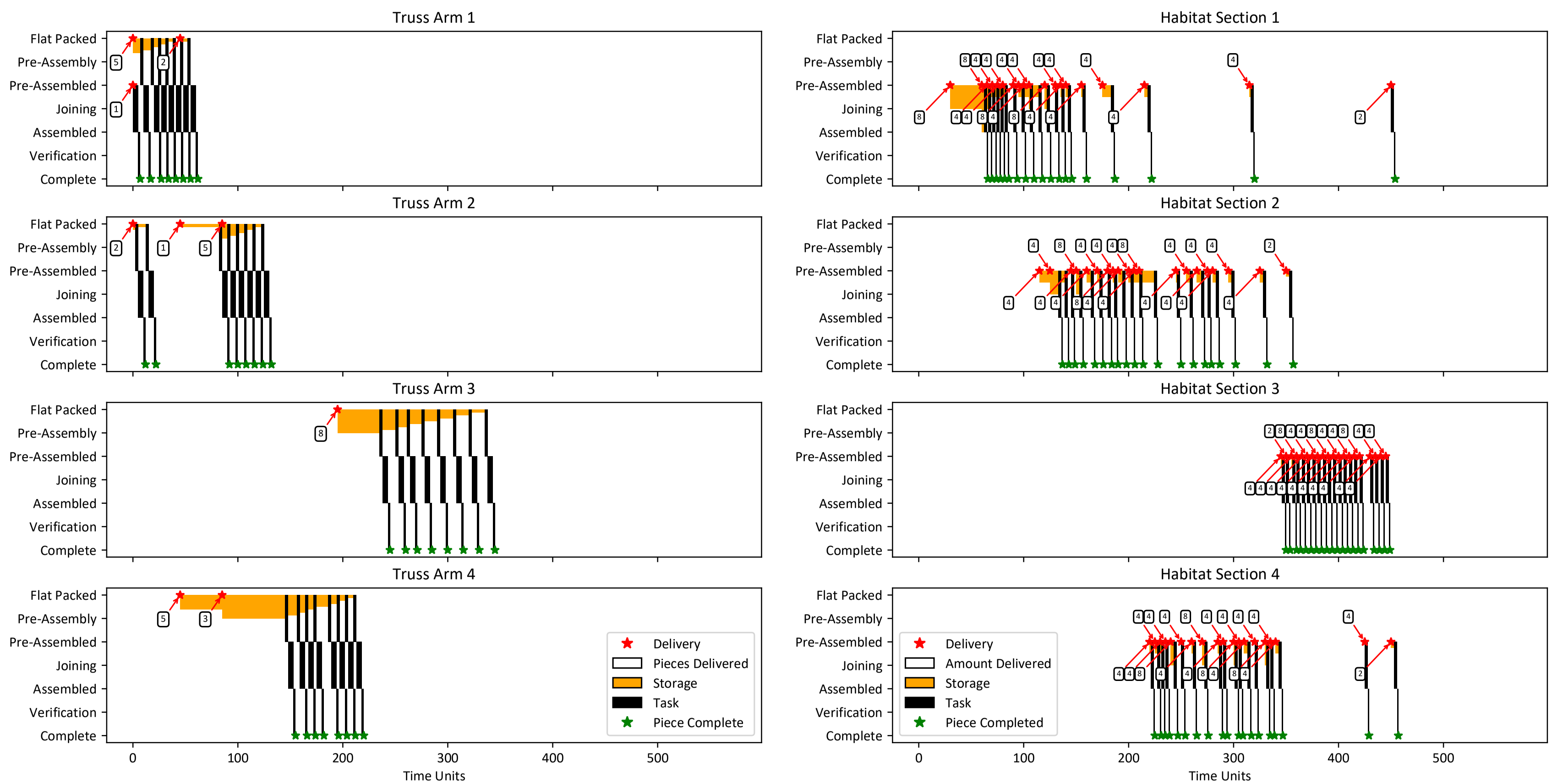

Figure 4.10: Results for the double storage capacity scenario under the time critical condition

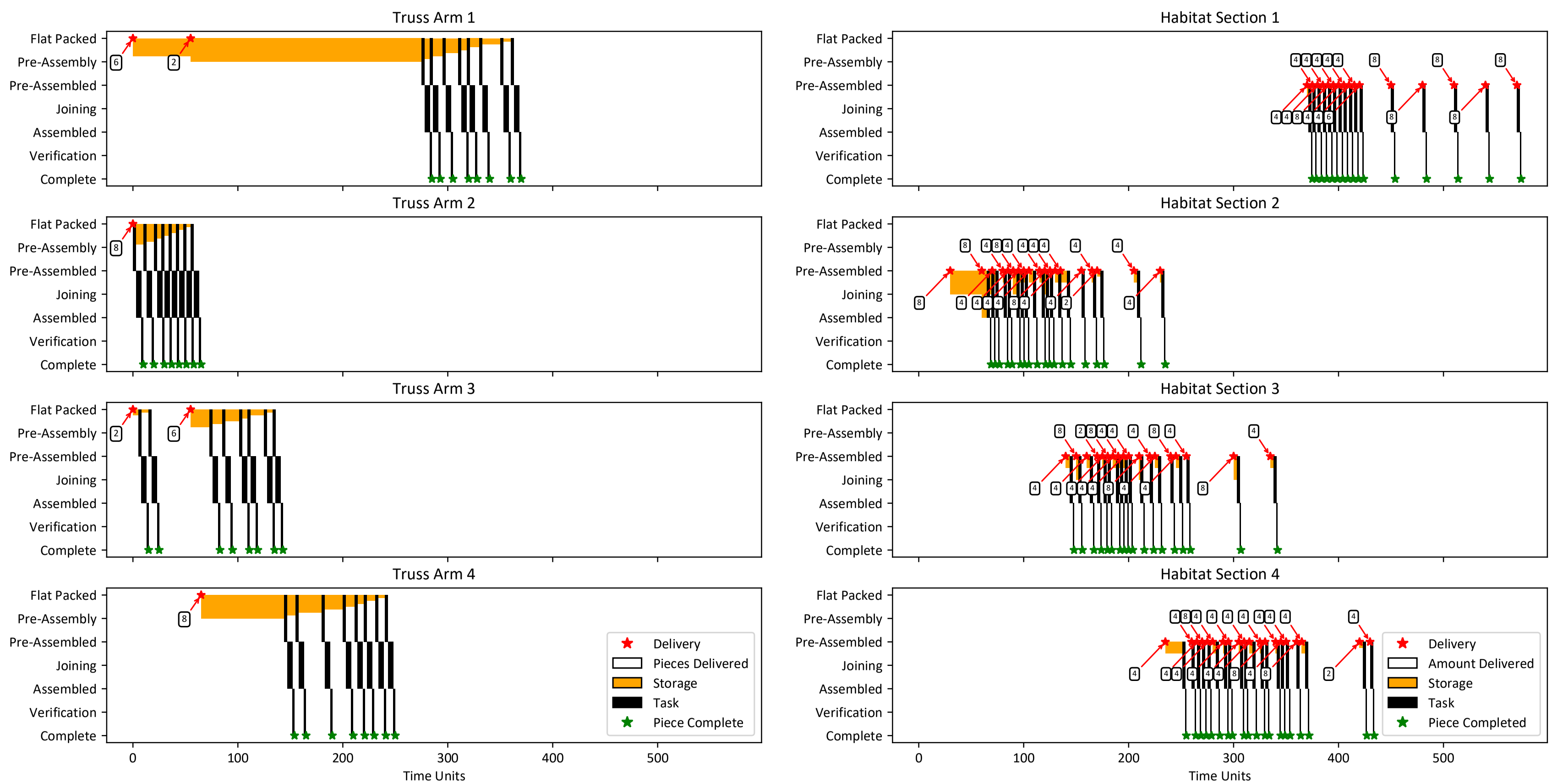

Figure 4.11: Results for the double storage capacity scenario under the cost critical condition

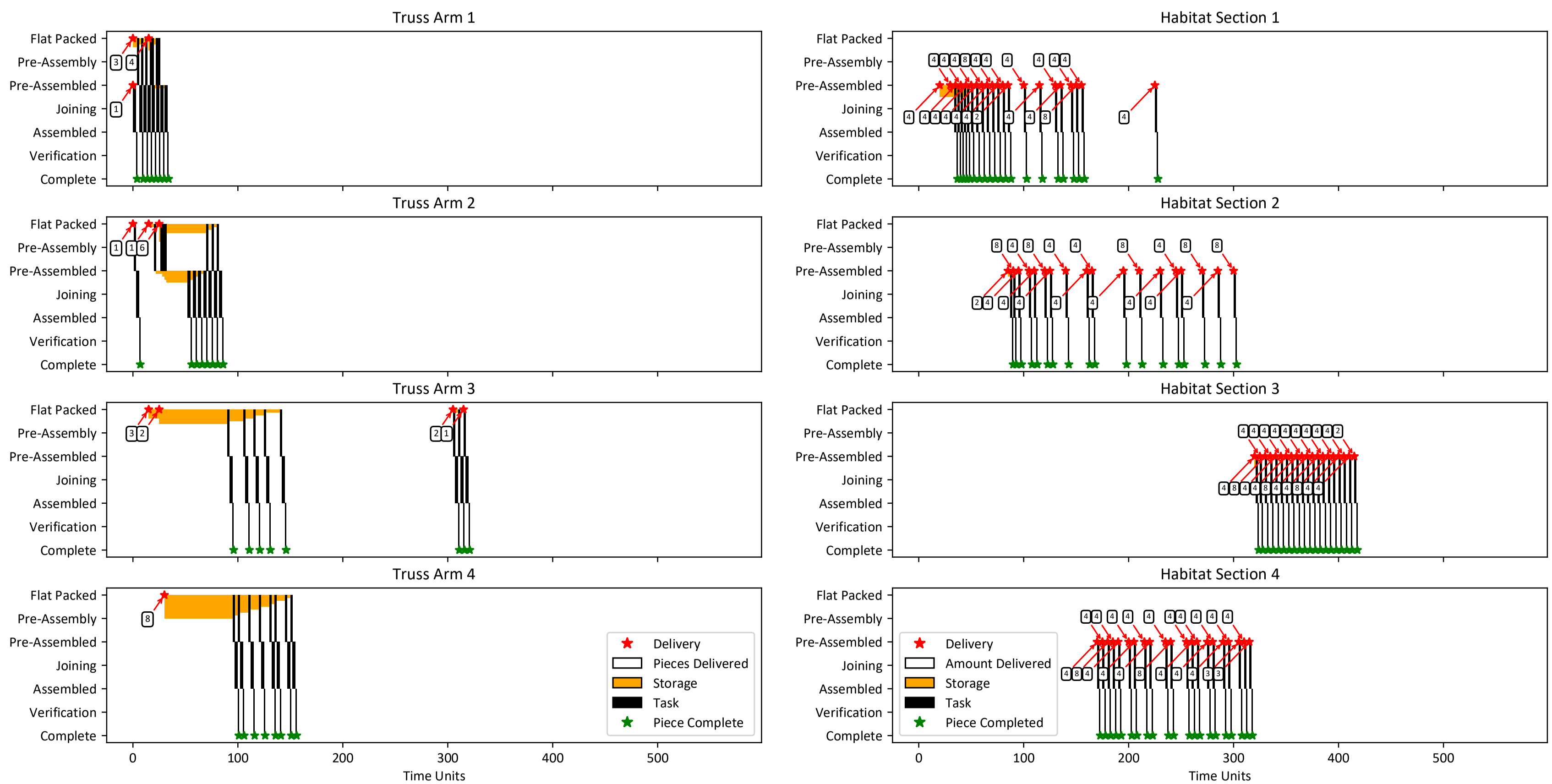

Figure 4.12: Results for the reduced duration scenario under the time critical condition

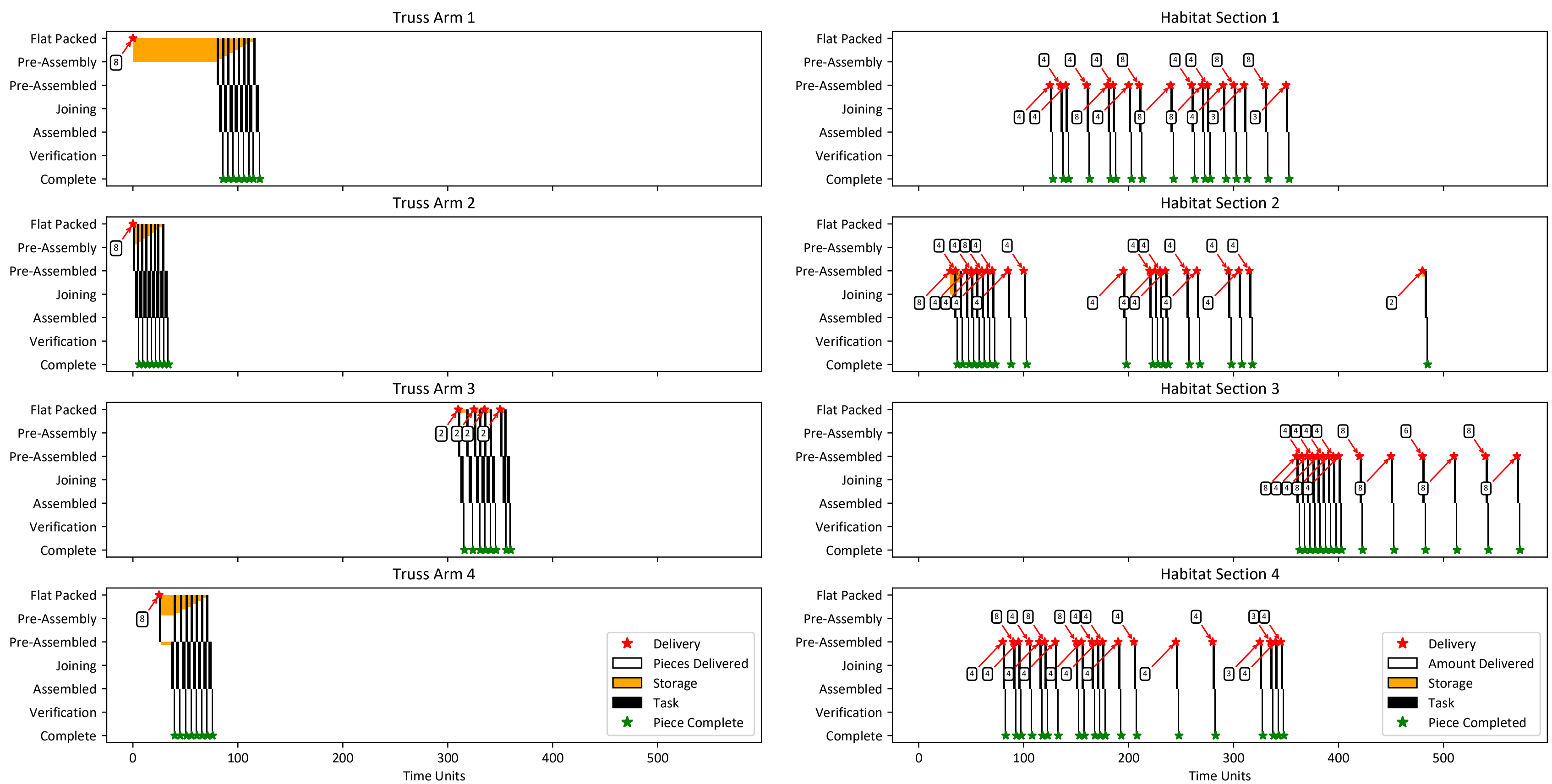

Figure 4.13: Results for the reduced duration scenario under the cost critical condition

once again similar results, shown in Figure 4.13. The AGSS completes assembly at 574 TU (4.7 years) using 74 deliveries, the same values obtained previously. Compared to the baseline, reducing the task duration has however a distinct effect in lowering the amount of ”storage time” used. Although habitat section deliveries are still spread out to maximize the use of the more cost effective higher capacity delivery type, the truss sections are all assembled within compact time spans, reducing the ”storage time” used on truss pieces.

Combined ISA Improvements Scenario under Time Critical Condition

This scenario combines the Double Task Capacity, Double Storage Capacity, and Reduced Duration scenarios in order to observe their joint effects. Figure 4.14 displays the results obtained under the Time Critical condition. The assembly is completed after 403 TU (3.3 years) using 81 deliveries. The process is 54 TU (162 days) faster than the baseline, and better than each individual scenario of improved ISA capability. Because the increased storage capacity is at no point being used, it can be concluded that this improvement has no effect even when combined with increased task capacity and reduced duration.

Figure 4.15 displays the results obtained under the Cost Critical condition for this combined scenario. The total assembly time is 572 TU (4.7 years) and uses 74 deliveries. There is once again, no improvement in the number of deliveries used, and a negligible improvement on the assembly duration over the baseline. However, the improvements to the total amount of ”storage time” used from previous scenarios are carried over and combined, resulting in very little storage necessary to achieve the same main metrics.

4.4.3 Improved Delivery Capability Scenarios

The impact of delivery capabilities was also assessed by increasing habitat delivery capacities and delivery frequencies separately. The values changed for these scenarios are shown in Table 4.3.

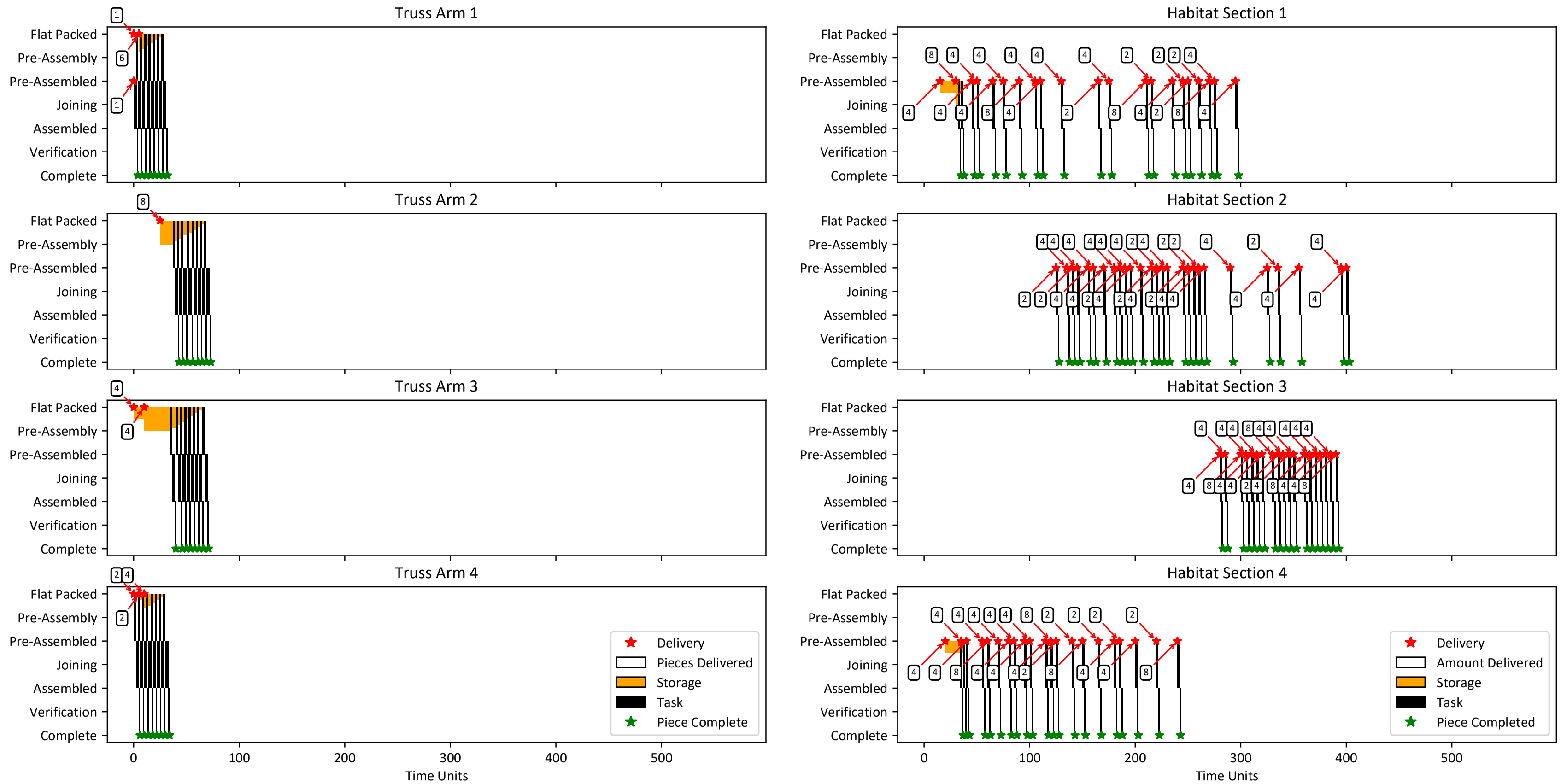

Figure 4.14: Results for the combined ISA improvements scenario under the time critical condition

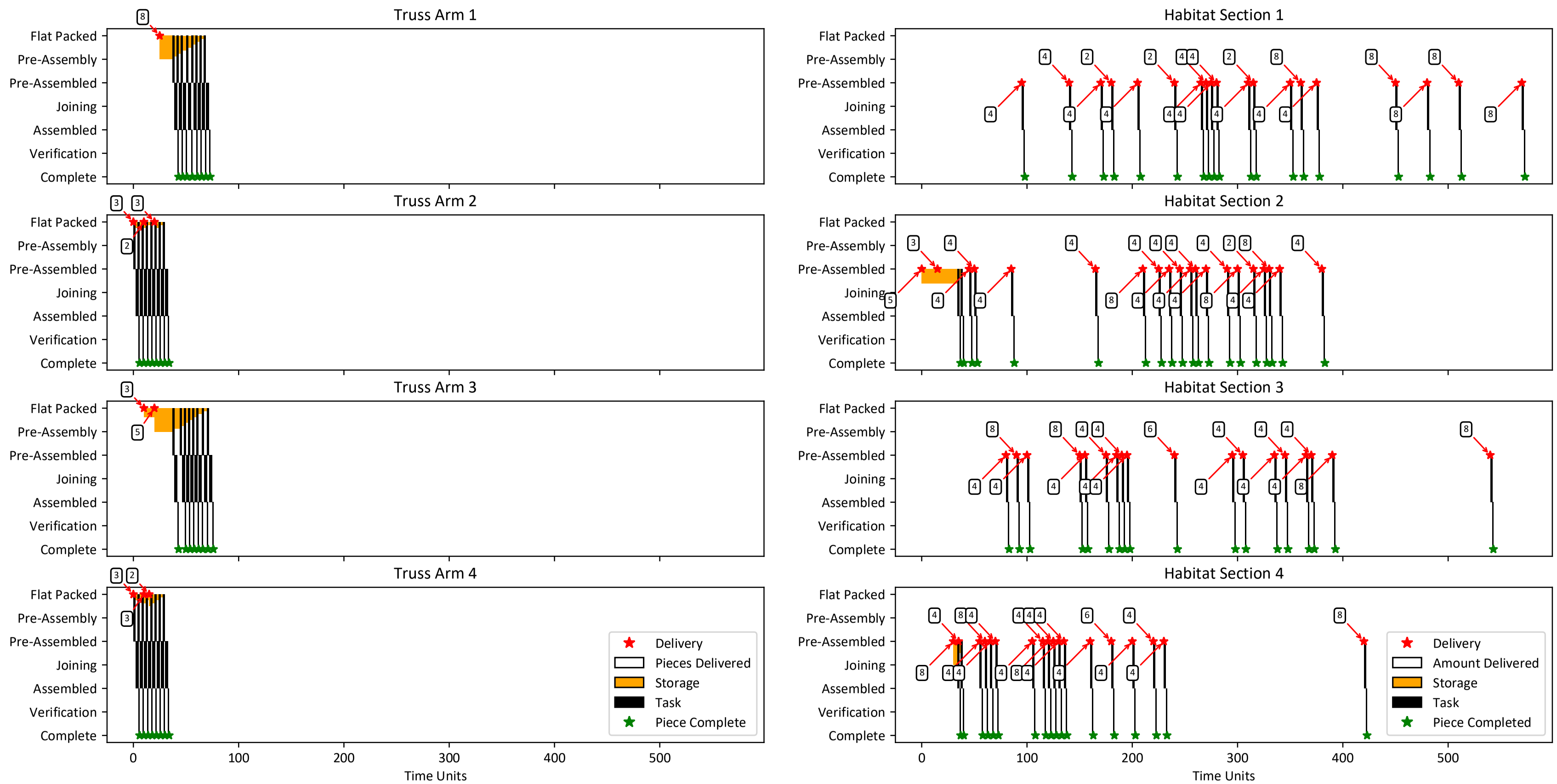

Figure 4.15: Results for the combined ISA improvements scenario under the cost critical condition

Table 4.3: Parameters for Advanced Delivery Capability Scenarios

Parameters	Higher Capacity	Higher Frequency
Delivery Types	2	2
Delivery Frequencies	1/5 and 1/30 TU	**1/4 and 1/20 TU**
Flat-Packed Truss Pieces	8 and 16	8 and 16 pieces
Pre-Assembled Truss Pieces	1 and 2 pieces	1 and 2 pieces
Hab Sections	**6 and 12 degrees**	4 and 8 degrees

Higher Delivery Capacity Scenario

In this scenario, the delivery capacity is increased for habitat pieces, for both delivery types, from 4 and 8 degrees maximum to 6 and 12, respectively. The results obtained under the Time Critical condition are shown in Figure 4.16. The larger habitat pieces being delivered result in a much faster assembly, completing after 339 TU (2.8 years) and using only 53 deliveries. This improvement is much higher than anything obtained by manipulating the ISA capabilities, exemplifying the importance of delivery capabilities in particular and specifically for habitat sections. Because the baseline of this scenario is primarily driven by delivery constraints, with assembly tasks completing before the next available delivery, larger pieces have a major impact.

The results obtained under the Cost Critical condition for this scenario are shown in Figure 4.17. As expected, the impact on the number of deliveries used is extreme, with only 44 deliveries used, or 30 less than the baseline. Despite this, with the objective of minimizing costs, the assembly completes in the same total time at 574 TU, to maximize the use of the more cost effective delivery type.

Higher Delivery Frequency Scenario

The availability of deliveries is increased in this scenario by adjusting the frequency of each delivery type, from one delivery every 5 and 30 TU to one every 4 and 20 TU, respectively. The results obtained for the Time Critical condition are shown in Figure 4.18.

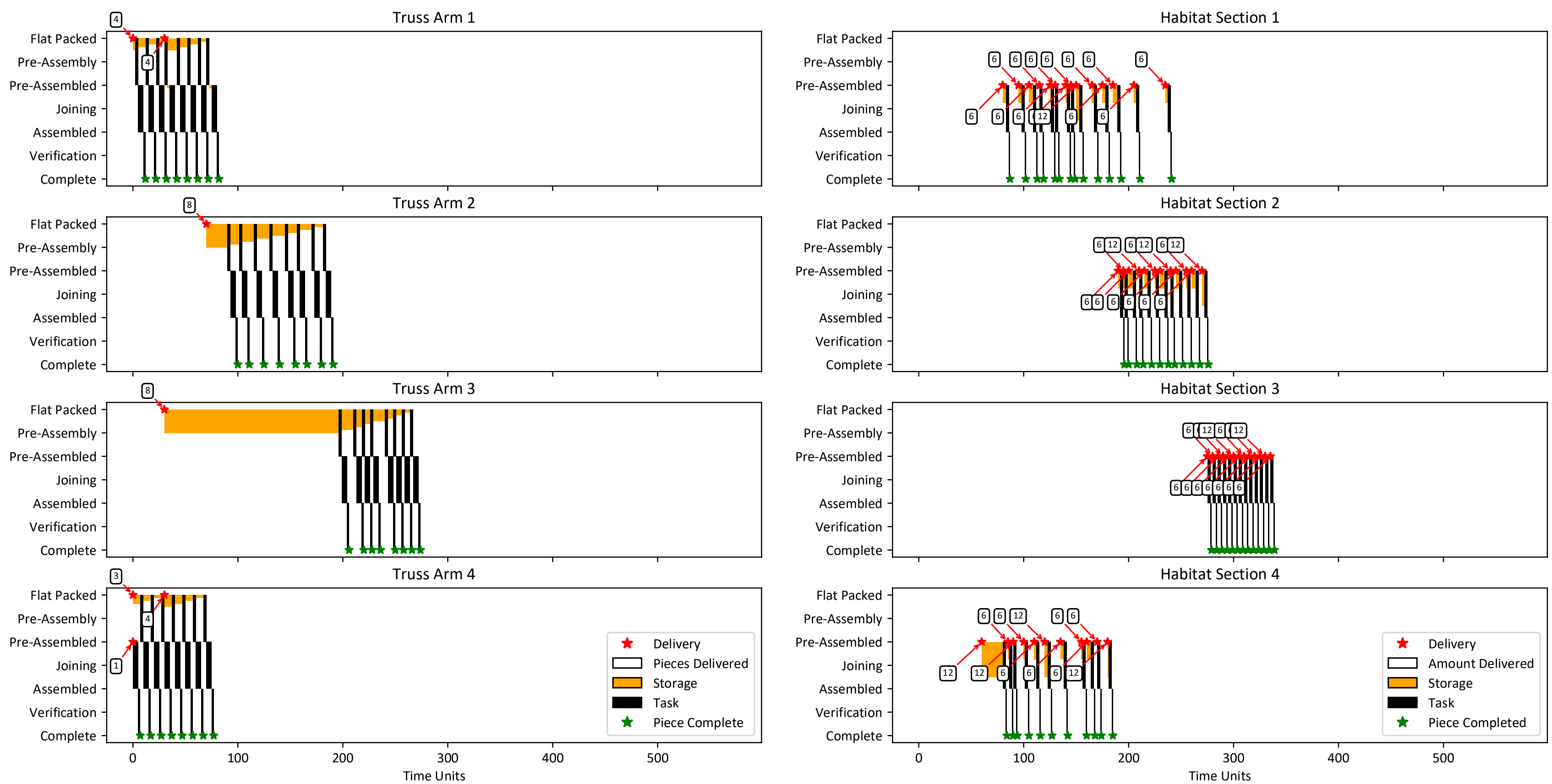

Figure 4.16: Results for the higher delivery capacity scenario under the time critical condition

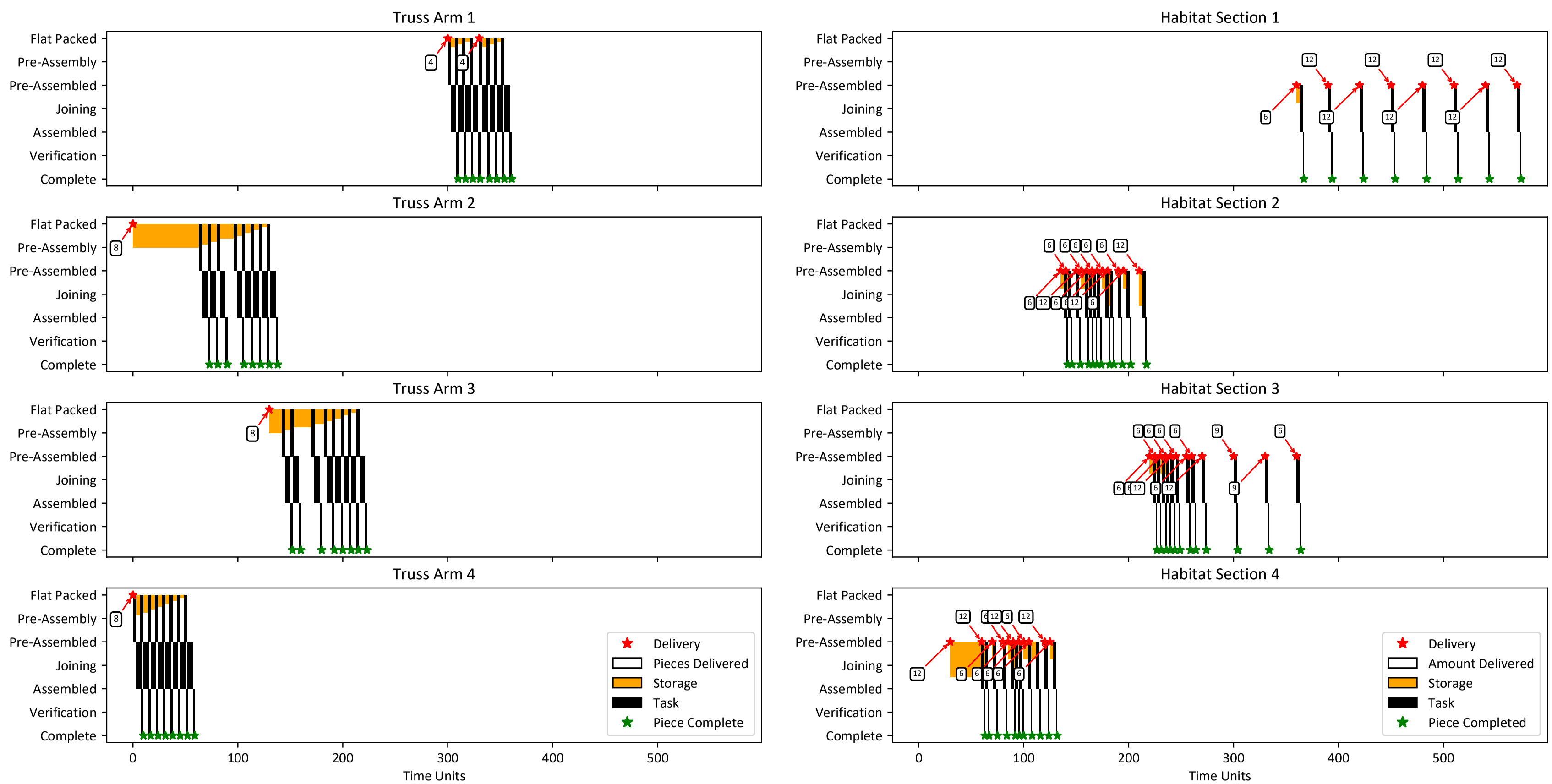

Figure 4.17: Results for the higher delivery capacity scenario under the cost critical condition

This increased frequency has virtually no impact on the assembly of the truss arms, which complete around 320 TU, similarly to the baseline. However, the habitat sections and the entire system are assembled overall faster, completing after 404 TU (3.3 years), 53 TU (5 months) faster than the baseline. As Figure 4.18 shows, the higher frequency available is exploited primarily by truss and habitat sections number 2, due to task duration capacity constraints. As a result, the effect is not as significant as increasing the delivery capacities.

Increasing the delivery frequency rather than capacity has two effects under the Cost Critical condition, as shown in Figure 4.19. The completion time is (surprisingly) delayed, from the baseline of 574 TU to 584 TU. In addition, the number of deliveries required is lowered to 65 from the baseline of 74. The reason is simple: with higher delivery frequencies, more higher capacity deliveries can be used, allowing the schedule to minimize cost by using these more efficient deliveries.

4.4.4 Additional Scenarios

Higher Delivery Frequency with Doubled Storage Capacity Scenario under Time Critical Condition

To identify if any benefits could be obtained from increasing the storage capacity for this sample case, the higher delivery frequency and double storage capacity scenarios were combined into one. The results obtained are shown in Figure 4.20. As can be seen, the increased storage capacity is only used on one occasion, for the storage of habit pieces of section 4. The assembly completes in 403 TU (3.3 years) using 76 deliveries, negligible improvements over the scenario with just higher delivery frequency. It seems that storage capacity has a very minimal impact on the assembly outcomes for this sample case, when the objective is to minimize the total time taken to construct the AGSS.

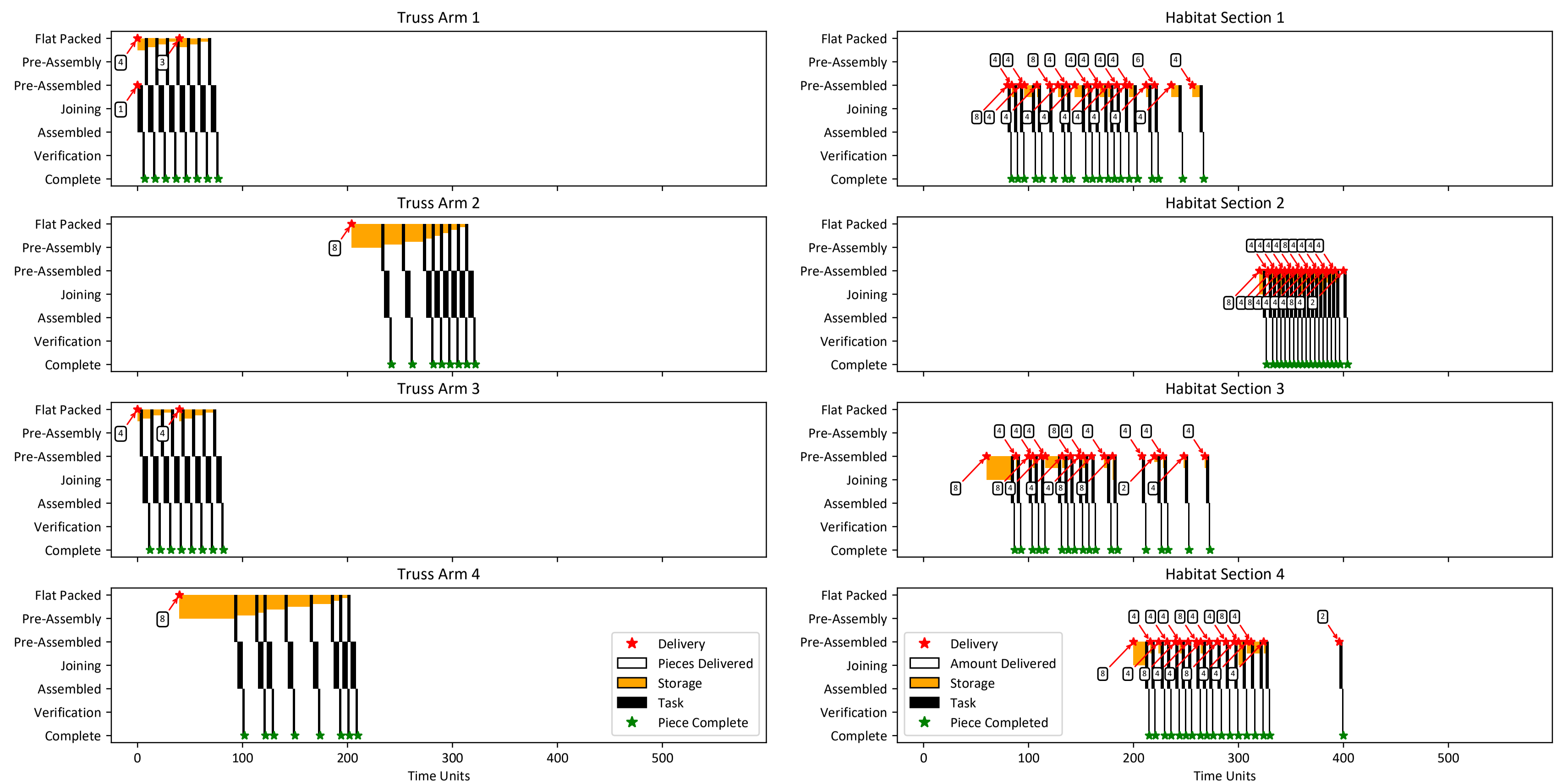

Figure 4.18: Results for the higher delivery frequency scenario under the time critical condition

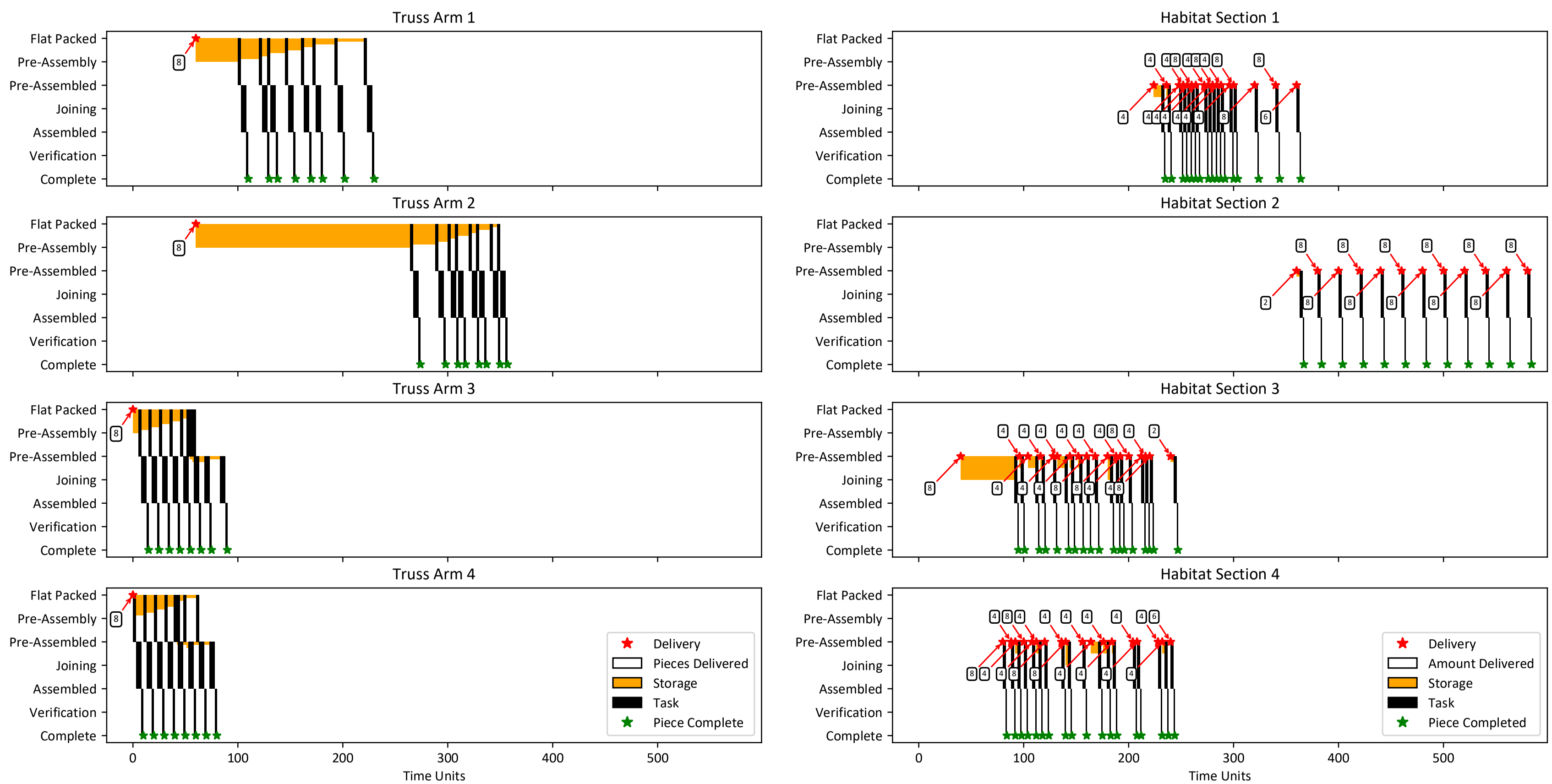

Figure 4.19: Results for the higher delivery frequency scenario under the cost critical condition

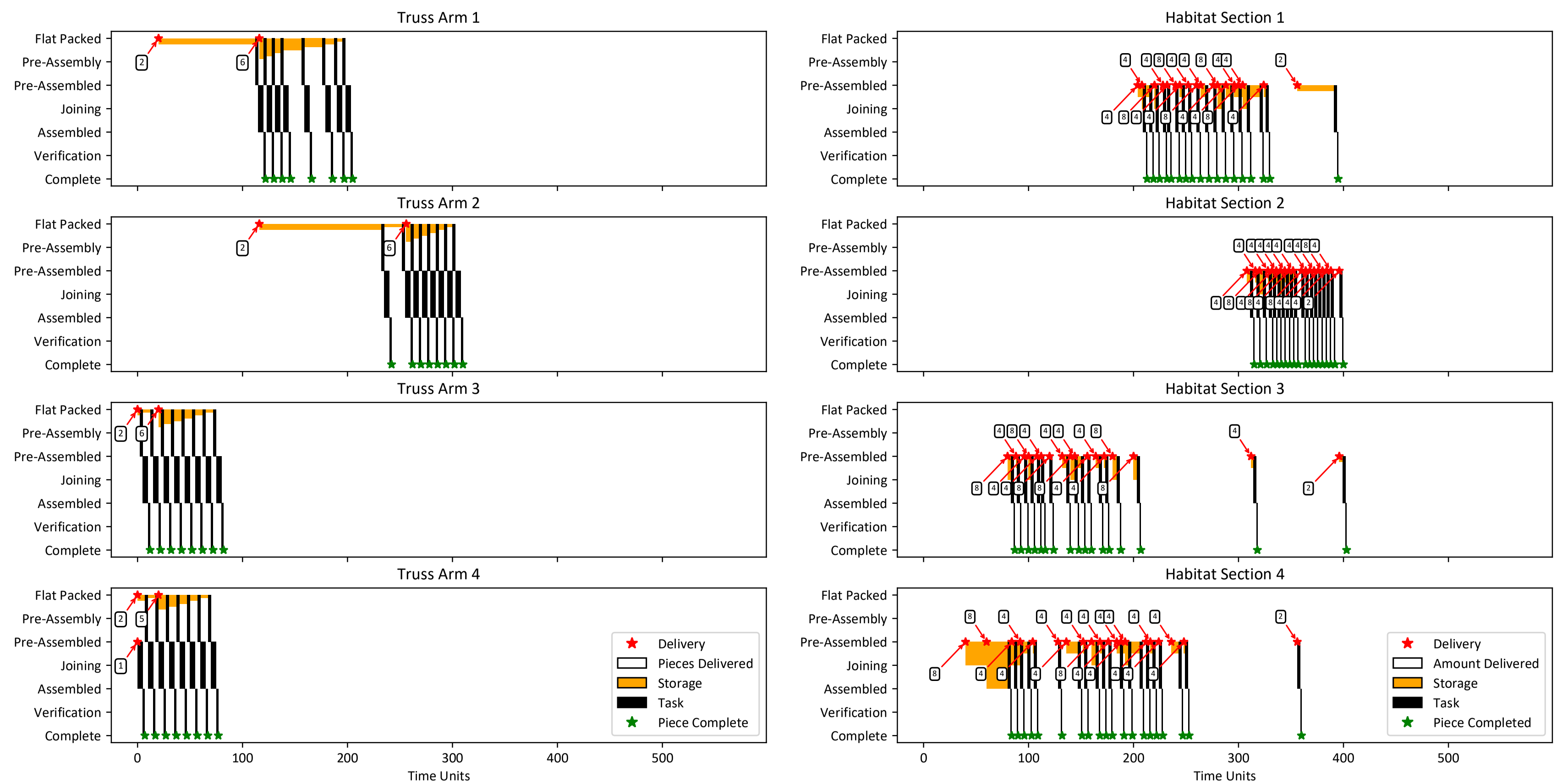

Figure 4.20: Results for the scenario combining higher delivery frequency and double storage under the time critical condition

Scenario with Alternative Task Durations

The results for the scenarios described above make very rare use of the ability to deliver truss pieces in a pre-assembled state. Although it is expected for the cost Critical Condition runs, Time Critical condition runs should make use of the ability to ”skip” an assembly task by delivering pieces in this more advanced state. A likely cause of this limited use is the baseline duration parameters for the pre-assembly and joining tasks. To observe under which conditions the use of pre-assembled pieces might be triggered, an additional scenario was run with modified task durations, shown in Table 4.4. These durations rearrange the total assembly time for truss pieces so that the portion needed for pre-assembly, the skippable step, is a larger fraction of the total.

Table 4.4: Parameters for Alternative Truss Assembly Duration Scenario

Parameters	Baseline	New Durations
Truss Pre-Assembly Task Duration	3 TU	6 TU
Truss Joining Task Duration	5 TU	3 TU
Truss VerificationTask Duration	2 TU	2 TU

The results obtained are shown in Figure 4.21. Compared to the baseline scenario and every other derivation, these new durations result in a total of 5 pieces being delivered in the pre-assembled state. Clearly, when the task durations cause different constraints, the model makes use of the ability to deliver pieces in advanced delivery configurations. Because the majority of the assembly of the AGSS is dominated by habitat section pieces, this degree of freedom has a minor impact.

4.5 Analysis of Results

4.5.1 Comparison of Time Critical Condition Scenarios

The Time Critical results obtained under the described scenarios are charted on a plot of Deliveries Used vs Completion Time in Figure 4.22. As can be seen, most results lie in the

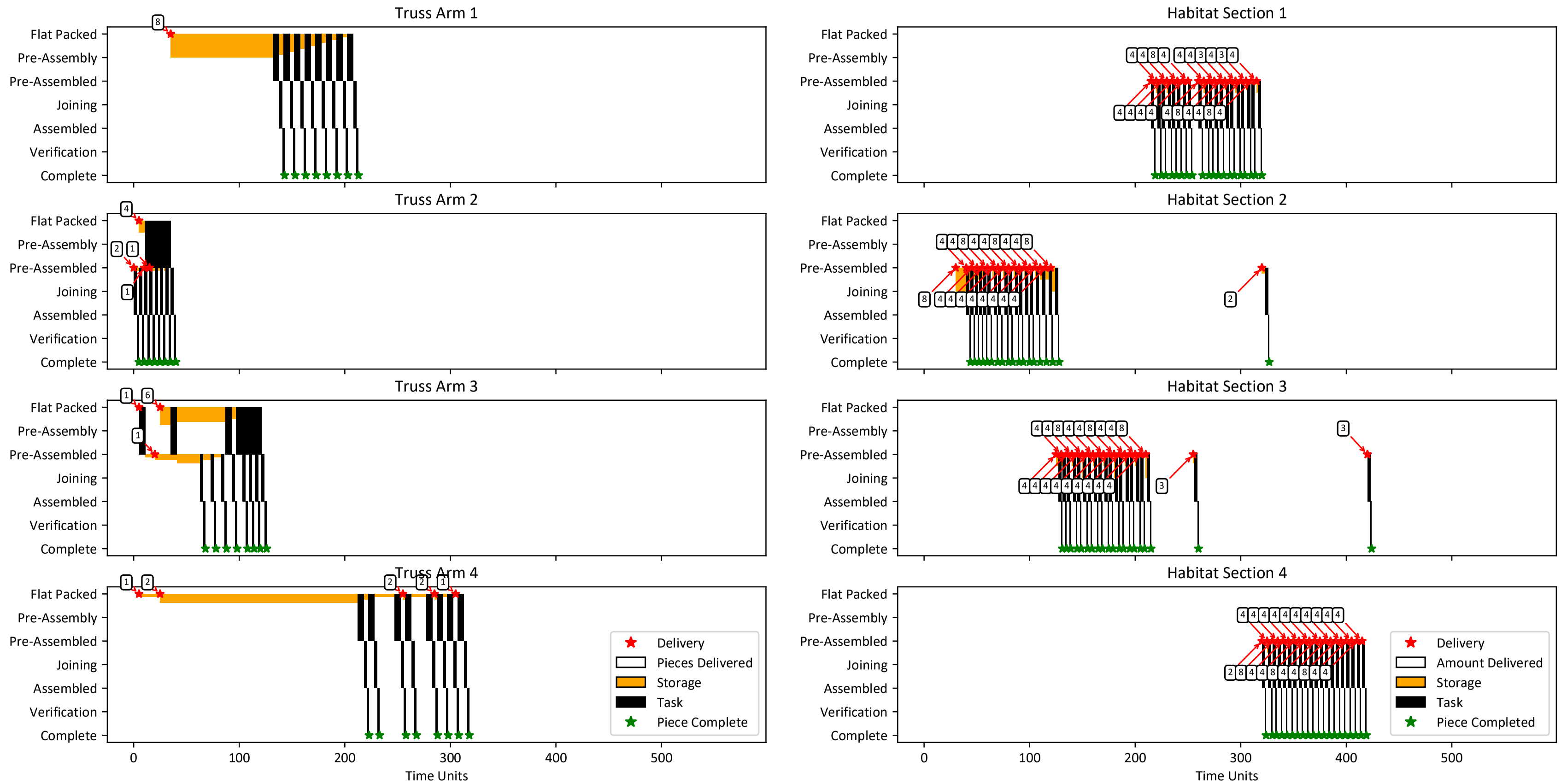

Figure 4.21: Results for the alternative task duration scenario under the time critical condition

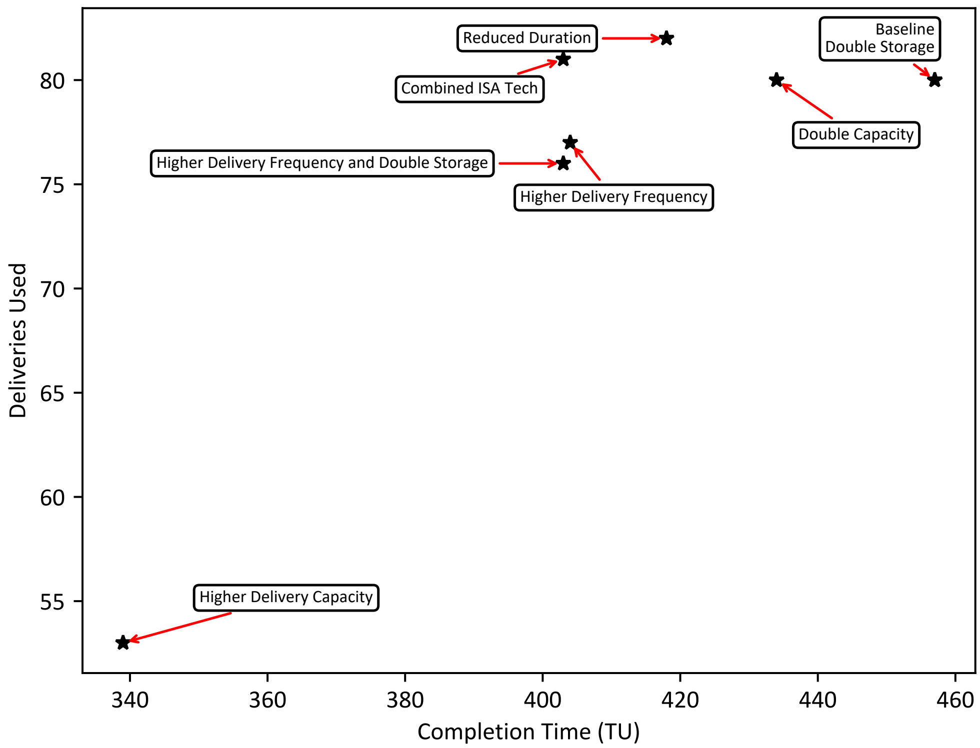

Figure 4.22: Comparison of deliveries used and completion time for different scenarios under the time critical condition

same region, with the scenario of higher delivery capacity for habitat pieces being the clear outlier. Doubling task capacity has a lower effect on completion time than reducing task duration. In turn, combining both improvements leads to a similar gain in time. The higher delivery frequency completes in approximately the same speed as the combined improvements of ISA technology. However, it is reasonable to assume that such an improvement would be vastly more costly than improving ISA. Additional storage capacity has a minimal effect on the the higher frequency scenario, indicated that it is clearly not limited by this parameter.

The scenarios can also be compared in terms of the total ”storage time” used. This metric can be calculated by adding the number of pieces stored at each time interval across the entire assembly process. It represents how much a scenario makes use of the available

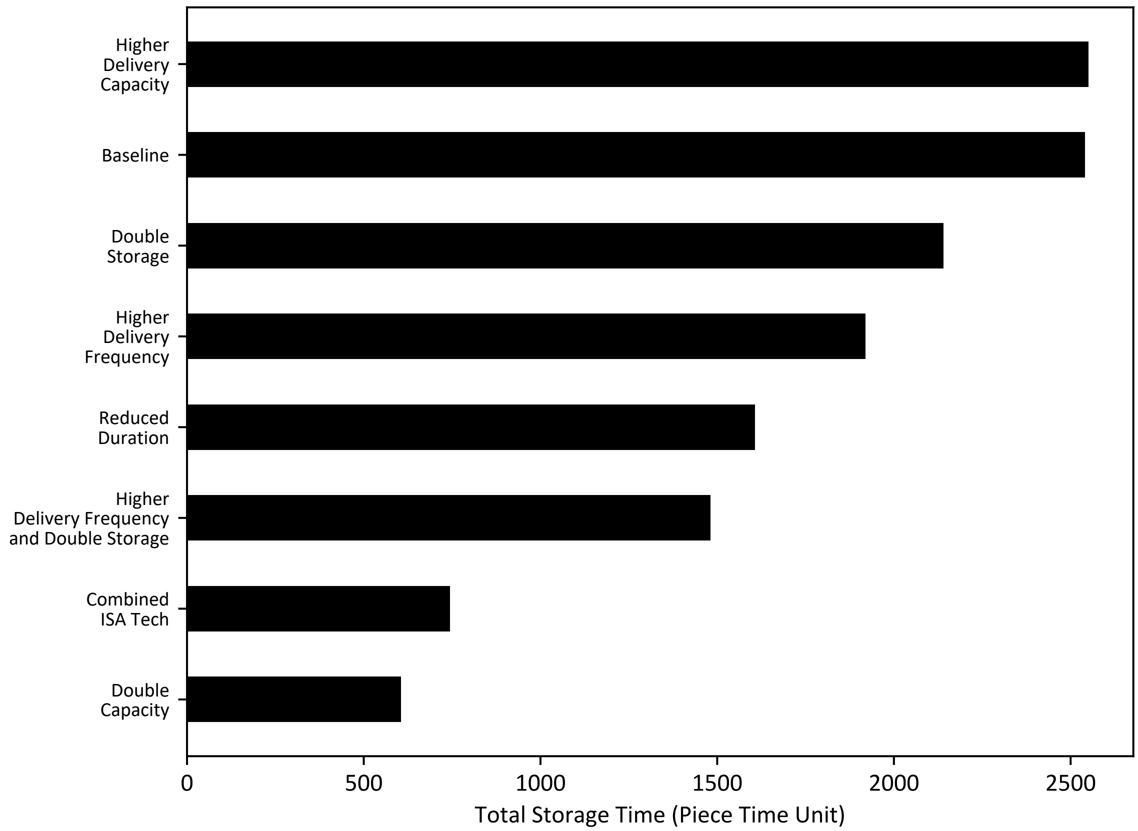

Figure 4.23: Storage use for each scenario under the time critical condition

storage capacity. The values obtained for each scenario are shown in Figure 4.23. The most important result observable is that the higher delivery capacity scenario, which outperforms every other scenario in terms of completion time and deliveries used, does so by using the largest amount of total storage time. On the other end of the spectrum, the double task capacity scenario uses the least amount of storage, despite being close to the baseline scenarios on the deliveries used vs completion time chart. These different assembly approaches can be clearly seen in their respective results, in Figures 4.8 and 4.16. With doubled task capacity, pieces are almost immediately assembled after delivery.

A final metric to compare the various scenarios is their delivery usage. The payload carried in each delivery event for each scenario is shown in Figure 4.24. Delivery A corresponds to the more frequent, lower capacity delivery type. Delivery B refers to the less frequent, higher capacity type. A few observations can be made from this figure. First, the

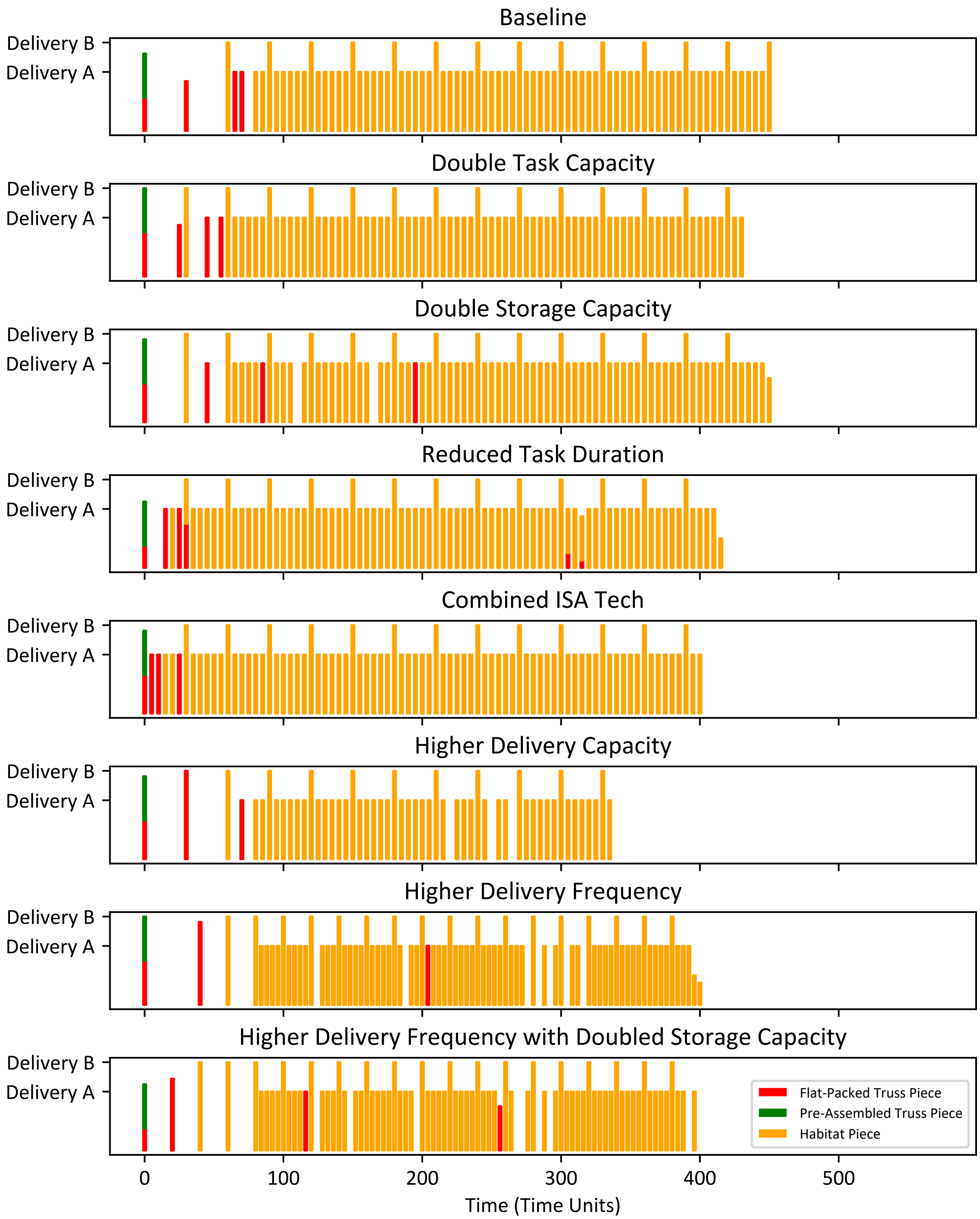

Figure 4.24: Delivery usage for each scenario under the time critical condition

majority of deliveries carry habitat pieces. Second, pre-assembled truss pieces are only ever delivered at the beginning of the assembly process. Finally, the results for the delivery scenarios and the double storage scenarios reveal some gaps characteristic of non-optimality. This indicates that better solutions could be achievable by dedicating more time to their computation.

These results reveal the relative impact of different factors on the overall assembly, and can allow trades to be made if provided with the cost of each improvement. Because reducing duration has nearly twice the impact as improving the task capacity, it might be better to focus on the performance of ISA "workers" rather than their amount. If affordable, improvements to delivery capabilities are vastly superior to those on ISA capabilities. However, their cost may be prohibitive. Increased storage capabilities have no impact on any factor when taken alone, and a minimal impact when combined with higher delivery frequencies. For the sample case of assembling the AGSS with a focus on completion time, increasing storage capacity has no benefit.

4.5.2 Comparison of Cost Critical Condition Scenarios

The metrics of Deliveries Used and Completion Time obtained for the various scenarios under Cost Critical conditions are plotted on Figure 4.25. As can be seen, only varying delivery capacity has a significant impact on the amount of deliveries used. No ISA capability parameter affected the two metrics plotted individually, while their combination resulted in a very small completion time improvement of 2 TU. Differentiating the various ISA capability scenarios requires the use of a different metric.

The "storage time" used for each scenario under the minimizing cost objective is displayed in Figure 4.26. As can be seen in the figure, despite performing equally in terms of completion time and deliveries used, these scenarios use storage very differently. Combining all ISA capability scenarios results in the least amount of storage used, while the doubled storage capacity scenario unsurprisingly uses the most.

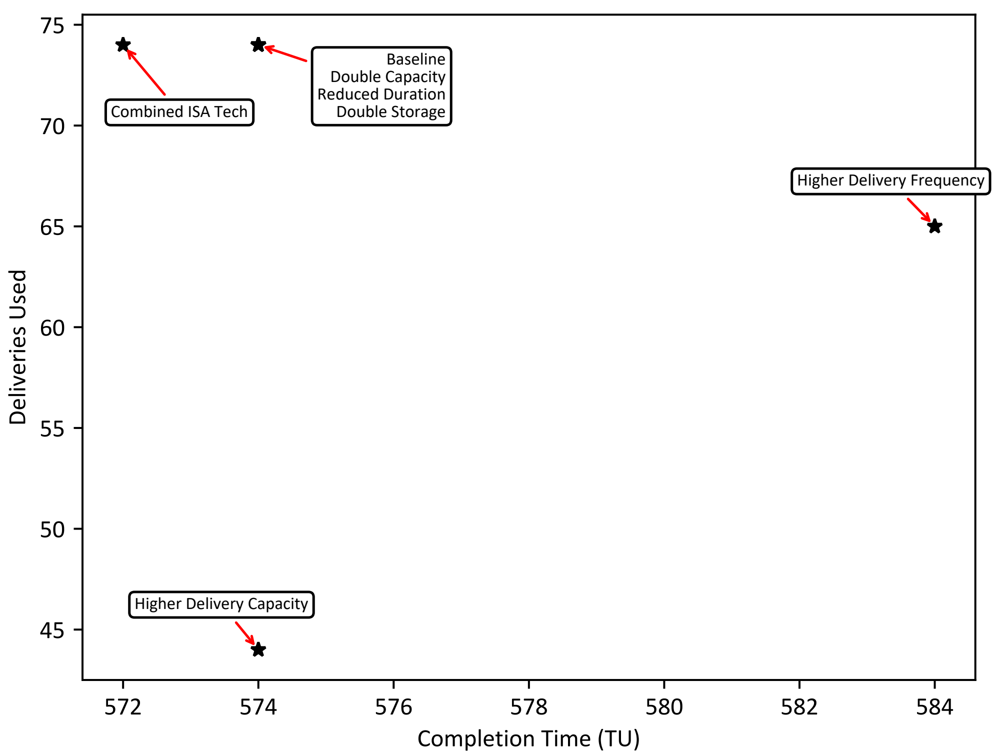

Figure 4.25: Comparison of deliveries used and completion time for different scenarios under the cost critical condition

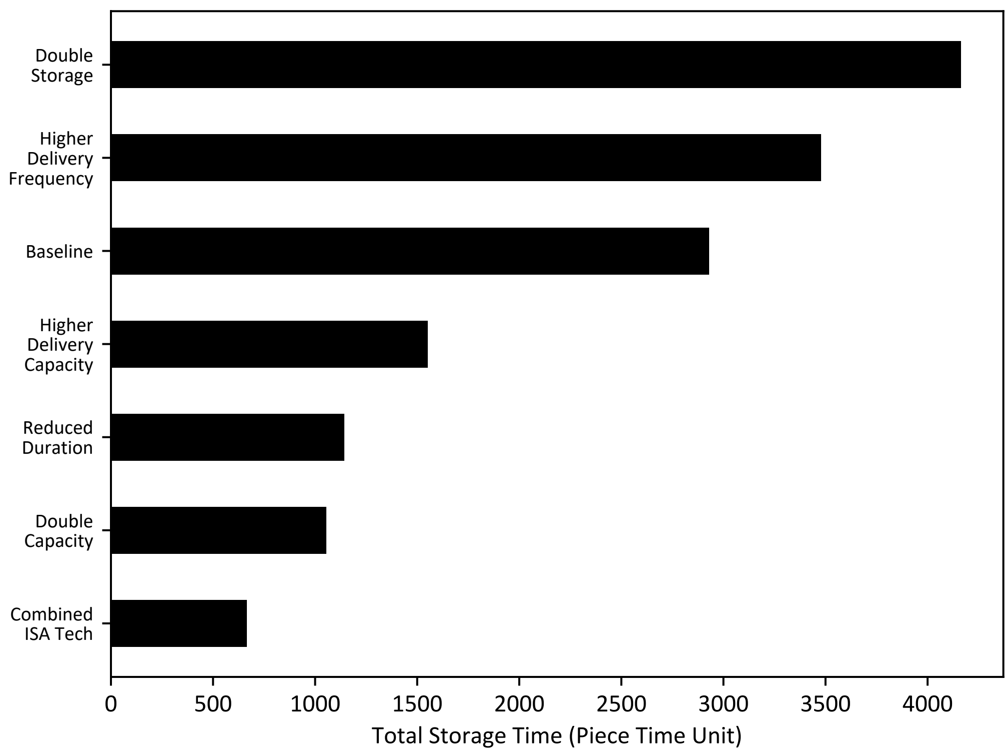

Figure 4.26: Storage use for each scenario under the cost critical condition

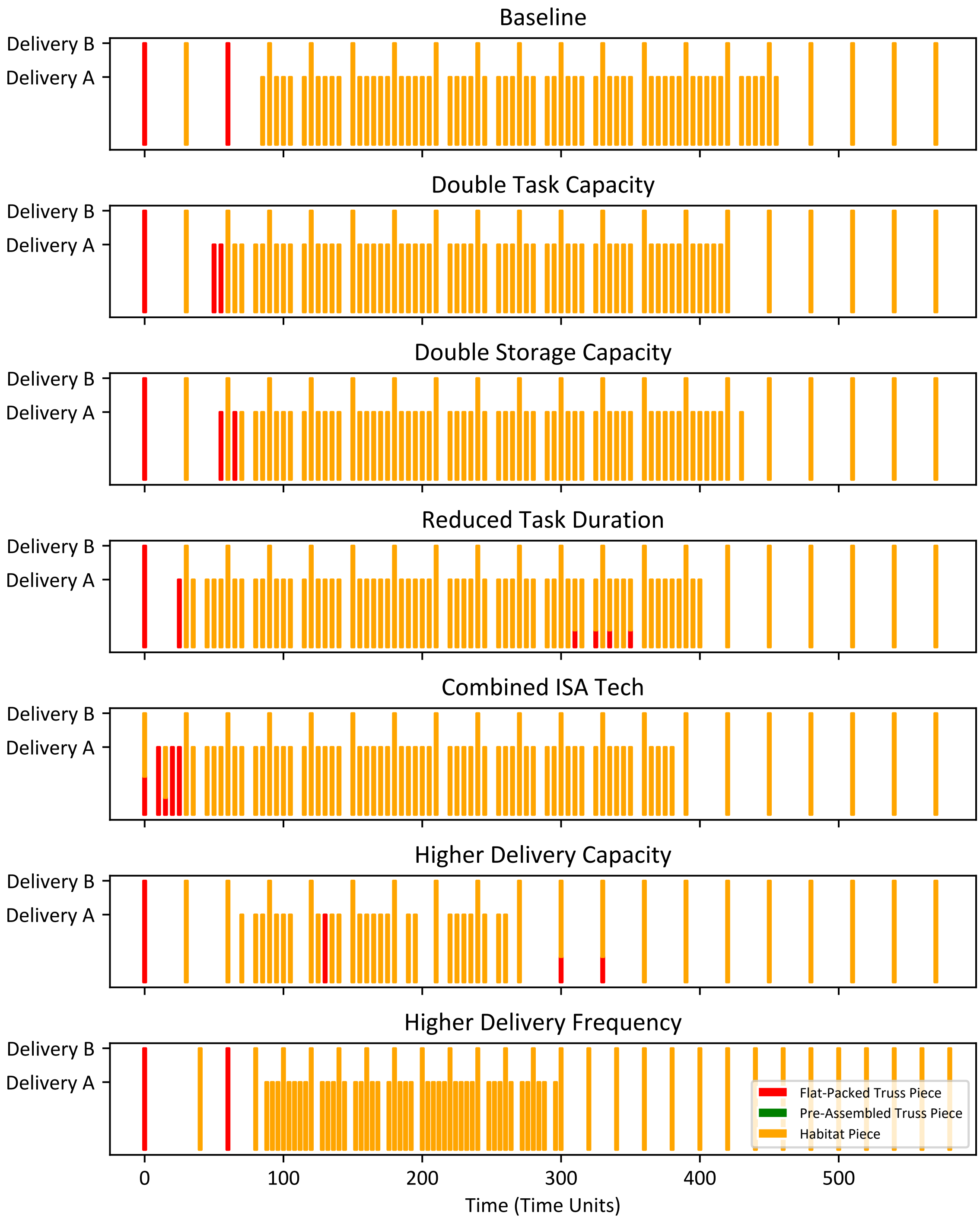

Figure 4.27: Delivery usage for each scenario under the cost critical condition

The payload carried by each delivery for these Cost Critical scenarios is plotted in Figure 4.27. As can be clearly seen, each scenario attempts to minimize costs by using a maximum amount of deliveries of type B. As a result, each improvement has only the effect of accelerating how early trusses can be completed, not the total amount of deliveries used. The increase in delivery capacity reduces the amount of deliveries of type A used. Each scenario is limited by the total time allowed for assembly, which limits the amount of type B deliveries used. If this time frame were extended, these scenarios would all make use of the type B deliveries exclusively. In order to balance time and delivery cost, an objective function relying on a common unit relating these two dimensions would have to be implemented.

4.5.3 Additional Discussion

The results obtained for each scenario under both Time and Cost Critical conditions reveal the major factors impacting the overall assembly process of the AGSS. When looking to assemble habitat sections as fast as possible in order to achieve an initial operational capability, improvements in ISA technology can reduce the assembly duration significantly, as well as reduce the total amount of storage time used. However, the ability to store more pieces at each state does not impact results significantly, even when combined with additional frequency. Increasing delivery frequencies has a comparable effect on the assembly time as taking all ISA technology improvements together. However, this improvement is not nearly as effective as increasing the capacity to deliver habitat pieces. Delivering a few additional degrees of habitat length has the most impact on the number of deliveries and the assembly time.

When the goal is to reduce delivery costs, scenarios attempt to maximize the use of more efficient delivery methods. As expected, increased delivery capacity has an extreme impact on this result. Increasing the delivery frequency has the surprising effect of slightly delaying the assembly completion. This minor effect is accompanied by a large increase in the cost effectiveness of the deliveries used. ISA technology improvements have little to no effect, but can reduce the amount of storage time used, in particular when combined together, which results in a quick assembly of trusses, allowing habitat sections to begin assembly early.

The use of different delivery configurations for truss pieces is limited by the dominance of habitat pieces over the entire assembly process, and by the balance of task duration between pre-assembly and joining tasks. When this balance is altered, pre-assembled pieces are delivered in higher number, but this trade remains a small portion of the overall assembly process. When assembling a system such as the AGSS, the effects of pre-assembly configuration, and ISA capability are overall small compared to anything that can affect the delivery of habitat pieces.

Of course, despite their effectiveness, improvements on delivery capability may be impossible to implement. Delivery systems used for the assembly of an AGSS may be sized with other missions and objectives in mind. Increasing frequencies beyond the already optimistic baseline of multiple launches per month may be unrealistic. In that case, ISA capability improvements may offer more realistic options for decreasing assembly time. To reduce costs, another option is to simply increase the total available time for the assembly of the system, allowing the exclusive use of higher capacity deliveries and advanced delivery configurations.

It is important to remember that these results are only valid for the specific case of the AGSS and the specific baseline chosen. A different balance between task duration and delivery frequencies could very well result in entirely different parameters becoming the most significant. Similarly, due to limited time, scenarios were only considered with improvements to the various capabilities: it may be that decreasing certain parameters has little or no effect, possibly offering savings when investing in certain capabilities.

CHAPTER 5
CONCLUSION

5.1 Summary and Conclusions

The success of future human space exploration missions will depend on their ability to overcome multiple technical and environmental challenges. Examples include the difficulty of landing human-scale cargo on Mars and the debilitating effects of long exposures to microgravity on the crew. Innovative designs, such as artificial gravity systems, can offer potential solutions to these challenges. These new designs are enabled by advanced ISA strategies, for which, in contrast to the currently used ISA strategies, more complex tasks are performed in orbit on a larger number of simpler pieces, which in turn can be delivered to orbit in more closely packed configurations. Technologies being developed today, in particular the robotic assembly of high-stiffness structures, will make these new ISA strategies available to mission designers, opening up new design trades. In these new design trades, the choices of how and when to deliver and assemble different system components are connected, and impact overall mission costs and durations. To inform the conceptual design of future missions, the **Research Objective** of this work was **to develop a methodology to conduct these new design trades**. To this end, the **Research Question** asked was:

What is the correct method of conducting design trades related to advanced In-Space Assembly strategies?

A review of relevant problems and methods identified common approaches as well as a gap in current capabilities. Designing a mission involving advanced ISA requires the optimization of the choice of delivery configuration, the arrangement and scheduling of deliveries, and the scheduling of assembly tasks. Although campaign logistics, space station logistics, and modular satellite assembly optimization models have been proposed individually, there has been no attempt to combine these various problems together. However, the use of network flow approaches for the formulation of linear programs was found to be common and effective in modeling space logistics problems, in particular for the optimization of campaign logistics. Therefore, an **Overarching Hypothesis** and a **Sub-Hypothesis** were formulated to answer the Research Question:

- **Overarching Hypothesis**: The correct method of conducting these new design trades tackles advanced ISA as a combined **logistics** problem. It uses a model of In-Space Assembly logistics as a whole, in which the delivery configuration, delivery schedule, and assembly sequence are degrees of freedom for the optimization of the entire assembly process.

- **Sub-Hypothesis**: An effective model of ISA logistics can be developed using a network flow approach. Such an approach can be adapted to the problem of ISA logistics to provide an optimal or near-optimal answer to the choices of assembly strategy, delivery schedule and assembly sequence.

The **Sub-Hypothesis** was first validated by successfully creating and testing a model of advanced ISA logistics using a network flow approach. Two types of time-expanded networks were developed to model different types of structural sections. A "fixed piece division" network models the assembly of structural sections consisting of a fixed number of pieces, and can consider multiple delivery configurations, tasks of different durations, and enforce 1-by-1 precedence relationships. A "variable piece division" network models the assembly of structural sections consisting of a variable of pieces of variable dimensions by tracking piece sizes and number, and can similarly enforce 1-by-1 precedence relationships. The assembly of complete space systems can be modeled using combinations of these networks related by precedence and capacity constraints. The effectiveness of this approach was confirmed by applying it to the case of an Artificial Gravity Space Station, modeled using 4 networks for its trusses and 4 networks for its habitat ring. The developed model allowed the fast computation of reliable results for the delivery and assembly sequences of an Artificial Gravity Space Station, validating the sub-hypothesis that a network flow formulation can succeed in modeling the complex problem of advanced ISA.

The **Overarching Hypothesis** was fully tested using the sample case of an Artificial Gravity Space Station. Using the developed logistics model, design trades were conducted at two levels. First, the optimization of the delivery configuration for each piece, and of the complete assembly and delivery schedule was obtained from the model under different scenarios and objective conditions. Trades between the choice of delivery type, the choice of delivery configuration, and other parameters were automatically performed by the logistics model to obtain an optimal results under a given scenario. Second, the importance of various parameters, such as the capacity to perform assembly tasks simultaneously, was revealed from the comparison of results obtained under different scenarios. As an example, the time and cost of the assembly of the Artificial Gravity Space Station were most sensitive to the delivery capabilities, and the delivery capacity in particular. Other parameters, such as the storage capacity, had little to no impact on the delivery costs and assembly time, but significantly affected delivery and storage use. By estimating the feasibility and viability of various improvements, trades involving the sizing of ISA and delivery capabilities were performed: although generating the most impact, improvements to delivery capabilities may not be feasible.

Using the developed methodology, both the optimal way to assemble a space system under fixed parameters, as well as valuable insights concerning which parameters might best improve the assembly process were obtained. The design trades conducted could not have been performed using separate models of delivery and assembly scheduling, as those degrees of freedom are not independent. The effectiveness of this combined logistics approach validated the **Overarching Hypothesis** proposed to answer the Research Question. By conducting design trades using the developed logistics model of ISA, the **Research Objective** of this thesis was attained.

5.2 Contributions

The main contribution of this work is a mathematical model of In-Space Assembly logistics. The mathematical model can calculate an optimal delivery schedule, assembly schedule, and payload manifest for the assembly of almost any space system. Furthermore, additional design trades can be conducted by varying delivery or assembly parameters and comparing the results obtained under different conditions.

The model assigns a flow network to each structural section composing a space system. These networks are related by additional constraints. Two network types were developed to model different types of structural sections. "Fixed piece division" networks model structural sections consisting of a fixed number of pieces of identical properties. "Variable piece division" networks model structural sections with a variable number of pieces of variable dimensions. When combined and linked by precedence and capacity constraints, these networks can together model the complete ISA logistics problem, while remaining solveable by standard processing units and commercial optimization solvers.

5.3 Future work

The work presented in this thesis can be further validated with the application of the developed methodology to other sample cases. In particular, systems consisting primarily of fixed division structural sections, such as EDL heat shields, may be more adapted to the application of the capability to optimize the choice of delivery configuration. Because the Artificial Gravity Space Station assembly is primarily driven by its large number of habitat pieces, this design trade was secondary in the sample case presented.

Additional time and computational resources can enable results with larger total time and finer time unit parameters than possible within the constraints of this thesis. In particular, solutions with the objective of minimizing costs can be optimized by allowing more total time for assembly. Finer time discretization can allow more distinction between various assembly processes. If a larger mathematical model is solvable, additional tasks can be modeled individually rather than lumped together into a handful of steps.

Finally, the model can be expanded to allow for more complex constraints and relationships. Modeling space systems as structural sections with pieces of similar parameters can restrict the capability to reflect real world problems. A new type of network may be needed to model structural section for which precedence relationships are more complex than 1-by-1 piece assembly. Similarly, additional constraints can be implemented to model new requirements for the assembly of space systems, such as day-night cycles for assembly tasks.

REFERENCES

[1] *Presidential memorandum on reinvigorating america's human space exploration program*, https://www.whitehouse.gov/presidential-actions/presidential-memorandum-reinvigorating-americas-human-space-exploration-program/, Accessed: 2018-06-26.

[2] S. Kelly, *Endurance A Year in Space, A Lifetime of Discovery.* 2017.

[3] G. Morgenthaler, "A cost trade-off model for on-orbit assembly logistics," in *4th Space Logistics Symposium*, 1991, p. 4148.

[4] S. A. Jefferies, D. C. Arney, C. A. Jones, F. Stillwagen, P. Chai, C. D. Hutchinson, M. Stafford, R. Moses, J. A. Dempsey, E. Rodgers, *et al.*, "In-space assembly capability assessment for potential human exploration and science applications," in *AIAA SPACE and Astronautics Forum and Exposition*, 2017, p. 5359.

[5] J. Lymer, W. R. Doggett, J. Dorsey, L. Bowman, A. Tadros, B. Hollenstein, B. King, K. Emerick, M. Hanson, and J. Boccio, "Commercial application of in-space assembly," in *AIAA SPACE 2016*, 2016, p. 5236.

[6] W. K. Belvin, W. R. Doggett, J. J. Watson, J. T. Dorsey, J. E. Warren, T. C. Jones, E. E. Komendera, T. Mann, and L. M. Bowman, "In-space structural assembly: Applications and technology," in *3rd AIAA Spacecraft Structures Conference*, 2016, p. 2163.

[7] J. Dorsey and J. Watson, "Space assembly of large structural system architectures (salssa)," in *AIAA SPACE 2016*, 2016, p. 5481.

[8] D. Hamill, L. Bowman, D. A. Gilman, and W. K. Belvin, "High leverage technologies for in-space assembly of complex structures," in *AIAA SPACE 2016*, 2016, p. 5397.

[9] *Definition of logistics*, https://en.oxforddictionaries.com/definition/logistics, Accessed: 2018-11-24.

[10] J. Snead, "Architecting rapid growth in space logistics capabilities," in *40th AIAA/ ASME/ SAE/ ASEE Joint Propulsion Conference and Exhibit*, 2004, p. 4068.

[11] X. Cheng, J. Guo, and N. Cui, "Space logistics development and future trend," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*, IEEE, 2009, pp. 2399–2403.

[12] T. Ishimatsu, O. L. de Weck, J. A. Hoffman, Y. Ohkami, and R. Shishko, "Generalized multicommodity network flow model for the earth–moon–mars logistics system," *Journal of Spacecraft and Rockets*, vol. 53, no. 1, pp. 25–38, 2015.

[13] C. Taylor, M. Song, D. Klabjan, O. de Weck, and D. Simchi-Levi, "A mathematical model for interplanetary logistics," *Logistics Spectrum*, vol. 41, no. 1, pp. 23–33, 2007.

[14] K. Ho, O. De Weck, J. Hoffman, and R. Shishko, "Dynamic network modeling for spaceflight logistics," in *AIAA Modeling and Simulation Technologies Conference*, 2014, p. 1491.

[15] K. Ho, O. L. de Weck, J. A. Hoffman, and R. Shishko, "Dynamic modeling and optimization for space logistics using time-expanded networks," *Acta Astronautica*, vol. 105, no. 2, pp. 428–443, 2014.

[16] ——, "Campaign-level dynamic network modelling for spaceflight logistics for the flexible path concept," *Acta Astronautica*, vol. 123, pp. 51–61, 2016.

[17] H. Chen and K. Ho, "Integrated space logistics mission planning and spacecraft design with mixed-integer nonlinear programming," *Journal of Spacecraft and Rockets*, pp. 1–17, 2017.

[18] K. Ho, J. Green, and O. De Weck, "Integrated framework for the design of crewed space habitats and their supporting logistics system," in *AIAA SPACE 2012 Conference & Exposition*, 2012, p. 5321.

[19] ——, "Improved concurrent optimization formulation of crewed space habitats and their supporting logistics systems," in *AIAA Space 2013 Conference and Exposition*, 2013, p. 5413.

[20] K. Ho, J. Green, and O. de Weck, "Concurrent design of scientific crewed space habitats and their supporting logistics system," *Journal of Spacecraft and Rockets*, vol. 51, no. 1, pp. 76–85, 2014.

[21] K.-P. Lin, Y.-Z. Luo, and G.-J. Tang, "Optimization of logistics strategies for long-duration space-station operation," *Journal of Spacecraft and Rockets*, vol. 51, no. 5, pp. 1709–1720, 2014.

[22] ——, "Optimization of space station's operational scenario parameters using physical programming," *Journal of Spacecraft and Rockets*, vol. 52, no. 2, pp. 629–633, 2015.

[23] K.-P. Lin, Y.-Z. Luo, J. Zhang, and G.-J. Tang, "Space station overall mission planning using decomposition approach," *Aerospace Science and Technology*, vol. 33, no. 1, pp. 26–39, 2014.

[24] G. Fasano, "A mixed integer linear programming model for traffic logistics management at the international space station," in *Modeling and Optimization in Space Engineering*, Springer, 2012, pp. 215–233.

[25] R. Moses, J. Van Laak, T. Chytka, R. Amundsen, K. Todd, D. Stambolian, S. Johnson, J. Dorsey, W. Doggett, J. Reeves, *et al.*, "Analysis of in-space assembly of modular systems," in *1st Space Exploration Conference: Continuing the Voyage of Discovery*, 2005, p. 2504.

[26] D. Sternberg, M. Chodas, C. Jewison, M. Jones, and O. De Weck, "Multidisciplinary system design optimization of on orbit satellite assembly architectures," in *Aerospace Conference, 2015 IEEE*, IEEE, 2015, pp. 1–14.

[27] E. L. Gralla and O. de Week, "On-orbit assembly strategies for next-generation space exploration," in *the 57th International Astronautical Congress. Valencia, Spain*, 2006.

[28] E. L. Gralla and O. L. De Weck, "Strategies for on-orbit assembly of modular spacecraft," *Journal of the British Interplanetary Society*, vol. 60, no. 6, pp. 219–227, 2007.

[29] J. Ahn, O. De Weck, and J. Hoffman, "An optimization framework for global planetary surface exploration campaigns," *Journal of the British Interplanetary Society*, vol. 61, no. 12, p. 487, 2008.

[30] G. Fasano, "Resource-constrained scheduling with non-constant capacity and non-regular activities," in *Space Engineering*, Springer, 2016, pp. 103–128.

[31] G. Fasano and M. C. Vola, "Space module on-board stowage optimization by exploiting empty container volumes," in *Modeling and Optimization in Space Engineering*, Springer, 2012, pp. 249–269.

[32] S. Gliozzi, A. Castellazzo, and G. Fasano, "Packing problems in space solved by cplex: An experimental analysis," in *Space Engineering*, Springer, 2016, pp. 129–150.

[33] P. T. Grogan, A. Siddiqi, and O. L. De Weck, "Matrix methods for optimal manifesting of multinode space exploration systems," *Journal of Spacecraft and Rockets*, vol. 48, no. 4, pp. 679–690, 2011.

[34] A. Siddiqi, O. L. De Weck, G. Y. Lee, and S. A. Shull, "Matrix modeling methods for spaceflight campaign logistics analysis," *Journal of Spacecraft and Rockets*, vol. 46, no. 5, pp. 1037–1048, 2009.

[35] K. V. Karottu, K. Hamada, A. Takezawa, M. Kitamura, and H. University, "Optimization of block division using nodal cut set method," *Ship Technology Research*, vol. 56, no. 3, pp. 142–152, 2009.

[36] M. A. Wibisono, K. Hamada, M. Kitamura, and K. V. Karottu, "Optimization system of block division using genetic algorithm and product model," *Journal of the Japan Society of Naval Architects and Ocean Engineers*, vol. 6, pp. 109–118, 2007.

[37] J. Kim, J. S. Park, T. Park, J. G. Shin, and K. H. Ko, "Optimum subdivision of curved hull plates for efficient ship manufacture using minimum strain energy," *International Journal of Computer Integrated Manufacturing*, vol. 25, no. 7, pp. 608–624, 2012.

[38] Z. Barbeau, S. Fehrenbach, and J. Jacob, "Development and testing of an inflatable artificial gravity system," in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 730.